FRACTAL REASONING

A Dissertation
Presented to
The Academic Faculty

by

Brian Keith McGreggor

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Interactive Computing, College of Computing

Georgia Institute of Technology
December, 2013

FRACTAL REASONING

Approved by:

Dr. Ashok Goel, Advisor
School of Interactive Computing,
College of Computing
Georgia Institute of Technology

Dr. Irfan Essa
School of Interactive Computing,
College of Computing
Georgia Institute of Technology

Dr. Gregory Abowd
School of Interactive Computing,
College of Computing
Georgia Institute of Technology

Dr. Andrea Thomaz
School of Interactive Computing,
College of Computing
Georgia Institute of Technology

Dr. Paul Corballis
Department of Psychology
University of Auckland

Date Approved: October 29, 2013

To my wife
Karen Michele Clark McGreggor,
to our children
Andrew Clark McGreggor and Sarah Grace McGreggor,
and to my parents,
Jimmie Lloyd McGreggor and Lanette Virginia Adams McGreggor.

Written on location in
Roswell, Georgia; Cape San Blas, Florida;
San Francisco, California; Seven Mile Beach, Grand Cayman;
Oxford, England; Edinburgh, Scotland;
Paris, France; London, England;
Washington, DC; San Antonio, Texas;
Destin, Florida; New Orleans, Louisiana;
Manchester, New Hampshire; Lowell, Massachusetts;
San Juan, Puerto Rico; Grand Case, St. Martin;
Houston, Texas; Harlem, New York;
and above the world during the all too many miles.

ACKNOWLEDGEMENTS

No one writes a dissertation alone, and I am no exception. A great many people over the course of my life have worked unwittingly to cause this work to be, and for me to be who I am now. There isn't space enough nor words sufficient to adequately thank them. But, I shall call out a few.

I wish to thank Ashok Goel, my advisor, for the many kindnesses and firm guidance he has shown me throughout the course of my research. To me, Ashok embodies the very essence of a great advisor, and my work has been shaped thoroughly by his consistent gentle drive. If there is someone whom I might hope to emulate in the coming years, it is certainly he.

I would like to thank the members of my committee, Gregory Abowd, Paul Corballis, Irfan Essa, and Andrea Thomaz, for their advice and counsel throughout the course of my PhD. Their collective wisdom has been invaluable to me, and I shall draw upon it repeatedly in the future.

Maithilee Kunda has been a lab mate, coauthor and a fellow journeyer throughout this trek. She introduced me to the Ravens, and we spent many hours discussing and debating the finer points of visual representations and reasoning. Thank you, my friend.

I’d also like to thank the members of the Design and Intelligence Lab. Bryan Wiltgen, David Joyner, Joshua Jones, Michael Helms, and Swaroop Vattam all befriended this prodigal PhD student, and I value each and every interaction and discussion we had through the years.

Stephen Fleming and my fellows in Georgia Tech's VentureLab have been unwavering in their support of my research. All the while that I pursued this degree, my day gig in VentureLab grew in complexity, and ultimately led me to become a lead instructor and curriculum advisor for the National Science Foundation's Innovation Corps program. For the latitude to chase all these big dreams simultaneously, I thank you.

My mom and dad, Lanette and Jim McGreggor, provided me with that great good foundation of life: a stable home, clear and unwavering support and encouragement, and the very best role models a son could ever have. I would not be here today if it were not for your love.

Andrew and Sarah McGreggor, you've watched your already geeky dad reach for these new heights with grace and good wishes. Thanks for being so understanding of those times I spent huddled over my keyboard, lost in thought, and for bringing me out of the deep zone when I'd been gone too long. You've brightened every one of my days, and I love you both.

Finally and most importantly, I would not have been able to come to this point in life's journey at all without the love and support of my best friend and wife of 31 years, Michele McGreggor. It is she who has made the keenest sacrifices, she who has listened to my faltering musings and wild thoughts, she who has brought me round when I became dispirited or discouraged, and she who has loved me beyond all expectations. I love you, Michele.

TABLE OF CONTENTS

Page

LIST OF TABLES AND CHARTS

LIST OF FIGURES

LIST OF ALGORITHMS

LIST OF SYMBOLS AND ABBREVIATIONS

RPM	Raven’s Progressive Matrices
SPM	Standard Progressive Matrices
APM	Advanced Progressive Matrices
CPM	Coloured Progressive Matrices
MAT	Miller’s Analogies Test
IFS	Iterated Function System

SUMMARY

Humans are experts at understanding what they see. Similarity and analogy play a significant role in making sense of the visual world by forming analogies to similar images encountered previously. Yet, while these acts of visual reasoning may be commonplace, the processes of visual analogy are not yet well understood.

In this dissertation, I investigate the utility of representing visual information in a fractal manner for computing visual similarity and analogy. In particular, I develop a computational technique of fractal reasoning for addressing problems of visual similarity and novelty. I illustrate the effectiveness of fractal reasoning on problems of visual similarity and analogy on the Raven's Progressive Matrices and Miller's Analogies tests of intelligence, problems of visual novelty and oddity on the Odd One Out test of intelligence, and problems of visual similarity and oddity on the Dehaene test of core geometric reasoning. I show that the performance of my computational model on these various tests is comparable to human performance.

Fractal reasoning provides a new method for computing answers to such problems. Specifically, I show that the choice of the level of abstraction of problem representation determines the degree to which an answer may be regarded as confident, and that that choice of abstraction may be controlled automatically by the algorithm as a means of seeking that confident answer. This emergence of ambiguity and its remedy via problem re-representation is afforded by the fractal representation. I also show how reasoning over sparse data (at coarse levels of abstraction) or homogeneous data (at finest

levels of abstraction) could both drive the automatic exclusion of certain levels of abstraction, as well as provide a signal to shift the analogical reasoning from consideration of simple analogies (such as analogies between pairs of objects) to more complex analogies (such as analogies among triplets, or larger groups of objects).

My dissertation also explores fractal reasoning in perception, including both biologically-inspired imprinting and bistable perception. In particular, it provides a computational explanation of bistable perception in the famous Necker cube problem that is directly tied to the process of determining a confident interpretation via re-representation.

Thus, my research makes two primary contributions to AI theories of visual similarity and analogy. The first contribution is the Extended Analogy By Recall (ABR*) algorithm, the computational technique for visual reasoning that automatically adjusts fractal representations to an appropriate level of abstraction. The second contribution is the fractal representation itself, a knowledge representation that add the notion of self-similarity and re-representation to analogy making.

CHAPTER 1

INTRODUCTION

We humans are expert at visual reasoning. We constantly receive a complex visual world, and interpret it: faces and figures, diagrams and paintings, landscapes and abstracts, all yield to our superimposed understanding. In the scenes we see, we recognize familiar objects, we notice novelty, and we are reminded of prior experiences. Yet, while the act of visual reasoning may be commonplace, how it is accomplished is unclear.

Though it is thoroughly influenced by prior research into human visual reasoning, the goal of my research and this dissertation is to develop a computational model of visual reasoning, and not a cognitive model per se. The model I propose herein is based on representing the received world in a fractal manner. Using this new representational lens, I illustrate the power and expressivity of the model in addressing problems of visual similarity and novelty.

In this introductory chapter, I begin with a few remarks concerning the inspiration I've taken from human visual reasoning, in particular familiarity, novelty, analogy, and abstraction. From there, I develop the problem statement, the research question, and what it is to construct a represented world. I note the several challenges to undertaking this work, and discuss the way in which I limit the scope. Next, I present the dissertation's thesis, and three hypotheses that it addresses. Finally, my work makes several significant contributions to science, and I conclude this introductory chapter with them, as a preamble for their further detailed discussion in the subsequent chapters.

The Inspiration of Human Visual Reasoning

My research has been significantly inspired by prior research into the way in which we reason about the visual world. In this section, I situate my research in the context of those sources of inspiration.

Novelty and Abstraction

Among the variety of processes, there are two aspects of human visual reasoning that seem especially powerful. These are the ability to notice novelty, and the ability to shift to an appropriate level of abstraction.

Novelty and Familiarity

Novelty and familiarity are related and intertwined (Sokolov, 1963): one might be very familiar with some visual object, yet may not consider it to be novel unless one encounters that object at time when least expected. As may be seen in Figure 1.1, novelty implies a context in which the visual signal is to be appraised; familiarity does not necessarily suggest this. Indeed, one may be entirely familiar, as an example, with what an apple looks like, but that apple would be unremarkable and lack novelty without some context.

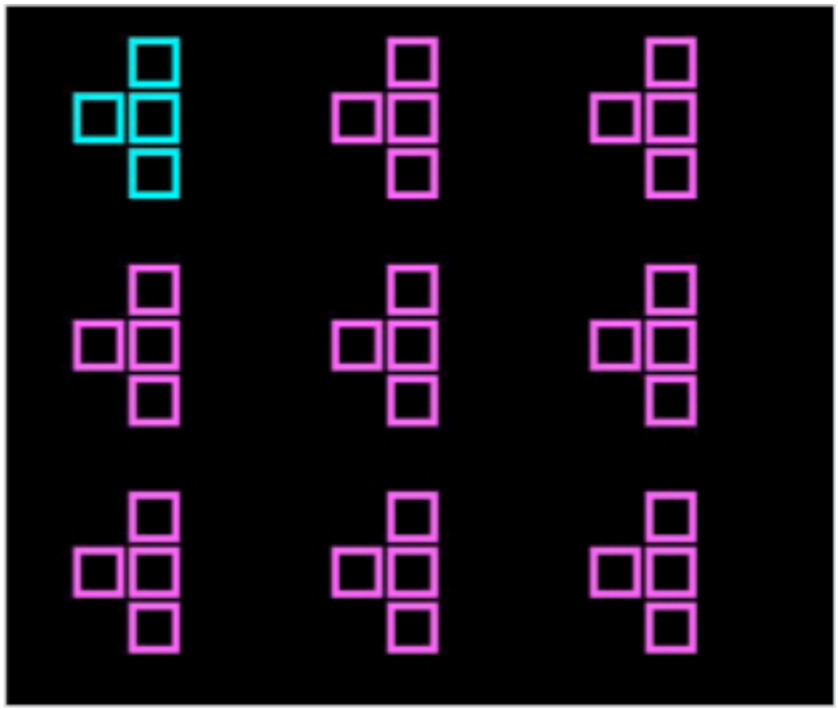


Figure 1.1. Novelty in Context.

In psychology, the phenomena of the orienting reflex (Sokolov, 1963) suggests that when stimuli are presented consistently, we (and all mammals) will turn or orient ourselves in the direction of a newly arriving distinct stimulus (Kishiyama & Yonelinas, 2003). Gradually, if the novel stimulus remains, we become habituated to it - it has become familiar. This habituation to stimuli is in a sense an outcome of a consistent perceptual state (Barsalou, 1999).

Perception and Memory

Perception and memory are connected via novelty as well. Several studies indicate that we recall novel events more readily than non-novel events (Hunt, 1995; Wallace, 1965), a phenomena known as the von Restorff effect (Von Restorff, 1933, as cited in Hunt, 1995). Hunt (1995) uses the term *distinctiveness* as a descriptive term to denote the perceptual saliency (or novelty) of events which violate the prevailing context, and argues that the novelty demands or attracts additional processing, perhaps through the mechanism of selective attention. This then causes further processing and elaboration, and the novel event is encoded into memory with the additional elaboration, which facilitates retrieval (Hunt, 1995).

Kishiyama and Yonelinas (2003) counter proposed, through their familiarity/ novelty hypothesis, that it is familiarity, and not recollection per se, that is sensitive to novelty. Their studies suggested that novelty affects both recollection and familiarity, but that recollection only exhibited the von Restorff effect if the stimulus was intentionally encoded, whereas familiarity exhibited the von Restorff effect when the stimulus was encoded intentionally or incidentally. Thus, Kishiyama and Yonelinas (2003) appeared to show that there may be two partially distinct responses to novelty, as measured by recollection. The key in their distinction, however, lay in the nature of the encoding of the stimulus.

Analogy and Representation

Our experiences provide a rich and ever changing context in which to situate, to compare, and to remember the in-falling visual world. This visual input may be novel, or it may be the same as our just-prior experience (and we would be habituated to it). This textural lexicon is the structure unto which is lain the newly arriving world for judgment.

It is in this contextualization of novelty that I find the bridge to that which Hofstadter views as the central core of cognition, the ability to make analogies (Hofstadter, 2001). Something regarded as familiar (or rather, similar, or analogous) must agree, in some sufficient number of aspects or ways or degrees, to that expectant tapestry of experience. For something to be novel, though, one need only note a single aspect or way or degree that doesn’t match.

In either case, it is that the something can regarded as either familiar or novel only when it is compared against some expectation, and that expectation comes from our experiences. Holyoak and Hummel (2001), in their description of the consensus of component processes involved in analogical thinking, specifically mention the retrieval of a source analog from long-term memory, in addition to others: a process for mapping that source analog to a target in working memory, the generalization and evaluation of inferences, and the induction of relational schemas.

Essential to these processes of analogy making is the representation (are the representations) upon which each operates. Most theories of analogy place particularly strong emphasis on structurally mapping relational presentations (Gentner, 1983; Holyoak & Hummel, 2001). Yet, all begin with that retrieval of an analog from memory, something with which to compare the present experience. The target stimulus, the just arrived experience, triggers retrieval, but what affords the retrieval?

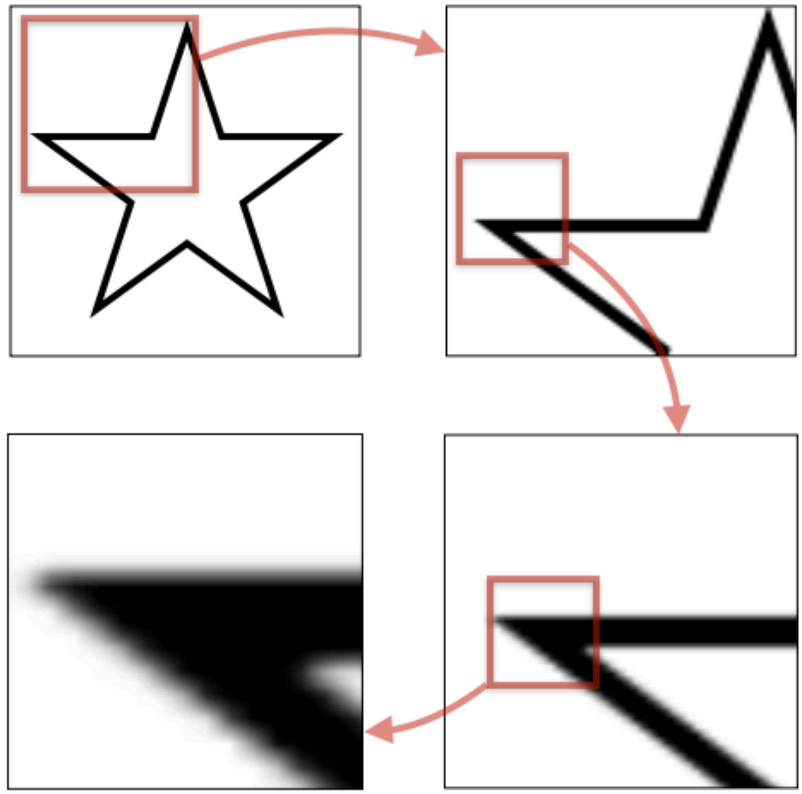


Figure 1.2. Successively zooming into detail.

Attention, Abstraction, and Representation

That we are able to notice novelty so quickly, to zone in on just that substantive difference, is remarkable. How is it, then, that we are able to make such swift shifts, and draw attention to those aspects of the visual world? In the prior section, I discussed the orienting reflex, and drew upon the work of Hunt (1995) to suggest the role that the process of attention plays in the elaboration and encoding of visual stimuli. Here, I expand on those remarks.

Signals and Attention

The detection of a signal, the onset of a stimulus, has been studied extensively in psychology. In these experiments, a signal is the stimulus being presented to a test subject, while noise is understood to be the rest of the environment (Goldstein, 2013).

Particular attention has been given to the awareness and report of the stimulus, through experiments which test the subject's ability to detect near-threshold signals (signals which are just distinguishable from the background noise) (Posner et al., 1980). The theory of signal detection (Green & Swets, 1966) makes the assumption that the observer is not-passive, but actively determines through some process whether or not to report the presence of a signal. Thus, in signal detection theory, not only is there the notion of sensitivity (the difficulty of distinguishing the signal from the background noise) but there is the notion of bias, the extent to which an answer (signal present/signal absent) is more probable. Both sensitivity and bias vary with the observer (Green & Swets, 1966).

The detection of a visual signal from a background, as one might expect in a task involving novelty, however differs from most of the experiments involving signal detection, as Posner et al. point out in a crucial way: the given signal is clearly above the noise threshold and 100% detectable (Posner et al., 1980). What is varying is the spatial position the signal occupies in the visual scene. This signal somehow attracts visual processing or attention, evoking a search of the scene for the signal itself.

Cognitive psychology offers at least two models for how visual attention shifts: the spotlight model and the zoom-lens model. In 1980, Posner et al. proposed the spotlight model of attention (Posner et al., 1980). The spotlight model describes attention as having a focus area of very high visual resolution, and a fringe area surrounding the focus but with a substantially lower visual resolution. The size of the spotlight, and the relative proportion sizes of the focus and fringe, are fixed. The zoom-lens model is the spotlight model, but relaxes the constraint of the sizes (Ericksen & St. James, 1986).

The tradeoff between these models rests in how much information is carried into the incoming signal, through the shifting in size of the region of high visual resolution. Both models maintain that the center of attention is wherever the geometric center of the focus area happens to land within the visual field. It's that last bit–the geometric center of the focus area–that poses an issue. How does one decide where to focus? Perhaps it is driven by some innate properties of objects in the visual field to focus on those areas: exogenous orienting is caused by stimuli in the visual periphery or an unusually bright or sharp contrast. Attention maybe driven endogenously, by what one is thinking at the time, by expectations about the scene, past experiences, or the task at hand, to direct the eyes to focus other places (Goldstein, 2013). It may be a bit of both (Berger et al., 2005).

Attention and Abstraction

It seems that one regards the entire image somehow, and then some further processing happens which directs the attention to focus on certain regions. When we receive a visual scene, high quality visual information is acquired only from a limited spatial region surrounding the center of gaze (the fovea of the eye): visual quality falls off rapidly and continuously from the center of gaze into a low-resolution visual surround (Henderson, 2003). Thus, the scene itself, for each gaze, varies in resolution: to obtain a more uniform resolution across the entirety of the scene requires the shifting of gaze, as illustrated in Figure 1.2. The visual scene then is available to be reinterpreted at a finer granularity in this manner. The degree to which the visual scene may be abstracted into finer or coarser resolution is mediated by some attention mechanism.

We humans effortlessly shift these levels of abstraction, changing the way in which the inbound visual world is modulated, all in the context and service of some task at hand. Yet we also are agents embedded in the world we receive, and have other means to modulate the abstraction level of the visual scene. We are able to move ourselves toward or away from a scene, or perhaps move some object closer to or further from us. In doing so, we alter the amount of the scene received by our eyes. As Wagemans et al. (2012a) point out, to deem something as novel involves the complex interaction of at least two relationships: the relationship between the observed and its context, and the relationship between the observed and the observer.

Binding, Integration and the Point of View

When we received the visual scene, we receive a great deal of information about the color, motion, and location. Treisman and Gelade (1980) proposed Feature Integration Theory as a two-stage process by which individual objects within a scene might be perceived, through the binding of features received at spatial locations to objects located there. To Treisman and Gelade, a received visual scene is first encoded in a variety of separable features or dimensions such as color, orientation, and brightness, in parallel across the entire scene. Then, through attending to various locations of the scene, these separable features at those locations are bound together, forming the perception of integral objects at those locations (Treisman & Gelade, 1980).

This perceptual binding of features in a scene into objects represents a deeper shift in perception: the shift from a received visual world to a collection of objects over which to reason. Marr (1982) proposed a similarly staged account of how human vision works.

Vision, according to Marr (1982), is a process that produces from images of the external world a description that is useful to the viewer and not cluttered with irrelevant information. To that end, Marr argues forcefully for the shift in point of view, from received image to representations of 3D models (Marr 1982). This progression of representations begins with features ala Treisman and Gelade, although Marr limits himself to considering only intensity. From an array of these intensities, a primal sketch representation is derived, consisting of lines and edges, boundaries and blobs. From the primal sketch, an analysis is made with respect to how the edges come together, and a 2½-D sketch is formed, with inferred surface orientations and depth information. Lastly, from the 2½-D sketch, a 3D model is made by inferring volumetric primitives and their arrangement in an object-centered coordinate frame.

The result is that through the vision process proposed by Marr, the received visual scene is transformed into a collection of representations of the objects contained within the scene. Other received features, such as color or motion, may be bound to these object representations, using feature integration theory, because the objects themselves occupy a spatial location within the scene. These object representations are then the stuff of further cognition (Marr, 1982).

Problem and Research

As I said in the preface of this chapter, my research and this dissertation are thoroughly influenced and motivated by studies of human visual reasoning. However, the focus of my research is on artificial intelligence, and the development of intelligent agents (Russell et al., 2010). To that end I would construct systems which could embody computationally some model of these processes of visual reasoning, the ability to notice novelty, to choose appropriate levels of abstraction, and make analogies over visual information.

Thus, the challenge posed would seem to distill such observations into the tractable and the computable. My body of work is marshaled toward addressing the following specific problem statement and research question.

The Problem Statement

Given that novelty and abstraction are so fundamental to visual reasoning, the problem lies in precisely how this may occur. This dissertation's problem statement is:

How might a visual scene be received to afford the notice of novelty at an appropriate level of abstraction?

The Research Question

My work is focused on the creation of computational models, and not expressly upon the delivery of a cognitively-plausible explanation of the phenomena. Therefore, the research question derived from the problem statement is restricted to computational models. Further, the question must be restricted toward problems which involve determining novelty or similarity, and in particular problems which involve analogy. Thus, this is the dissertation's research question:

How might a cognitively-inspired computational model receive a visual scene and, in the service of some visual analogical task, notice novelty at an appropriate level of abstraction?

In the next sections, I will expand upon what I mean by the visual scene and visual tasks, to motivate my thesis statement and its attendant hypotheses.

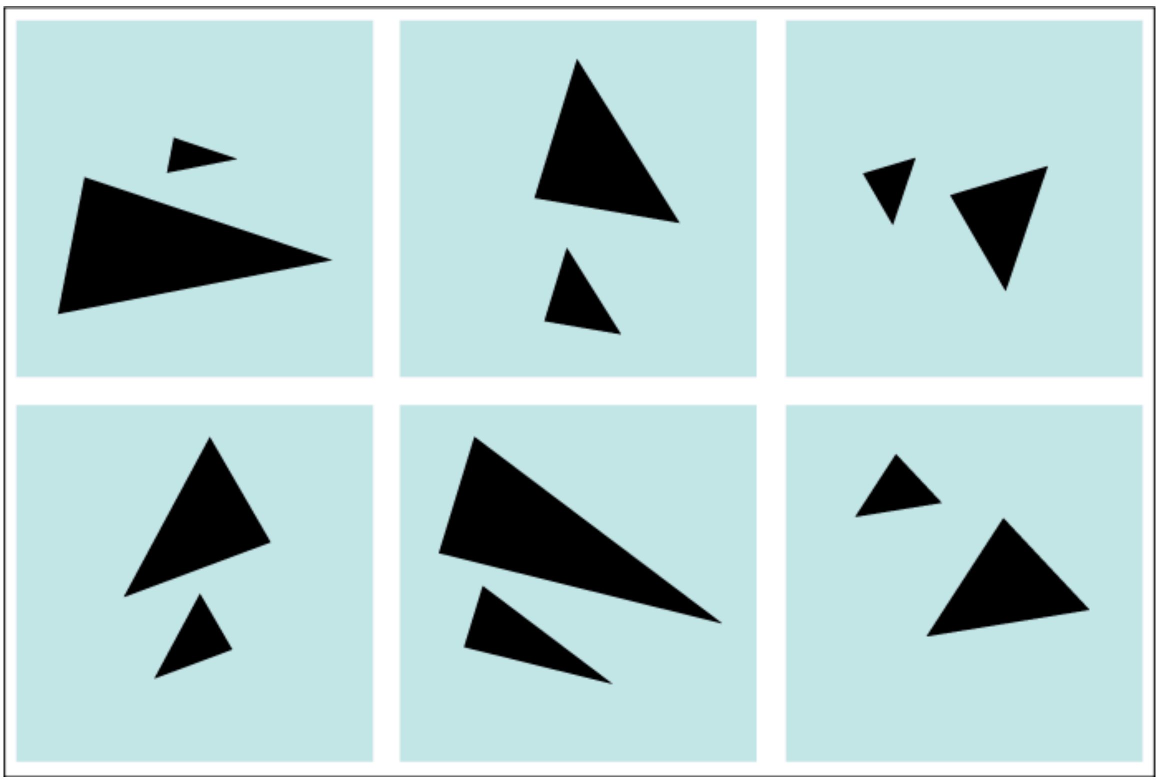


Figure 1.3. A 2D visual scene

Receiving a Visual Scene

Simply put, for humans, to receive a visual scene is to gaze upon it, and receive light information into the eyes. Computationally, it is analogous: a scene is received once it is input in some format. Figure 1.3 shows one such scene.

However, it is important to draw a distinction between what the world is and what the world affords. Some object in the world may be labelled as novel by a particular observer, but that is not sufficient to suggest that the object in question would be regarded as novel by every plausible observer. Novelty depends upon context, and every observer's context–her internal, perceptual context–will vary.

Similarly, while a visual scene arriving from the world is continuous, it does not directly offer a notion of abstraction, merely offering an opportunity for an observer to receive the world in differing manners through some enaction of the observer upon or within the world (changing the nature of the light which falls upon an object or manipulating the object somehow) or through some modification of the observer as an entity within the world (moving closer or further to an object, or changing the visual system mechanically via squinting, and the like).

Requirements

The acts of noting novelty and shifting abstraction are cognitive acts which occur entirely within the mind of a human observer. The visual scene of the world itself affords them, but it is the observer performs them. That is, some set of cognitive processes occurs within the observer to accomplish these feats.

As the goal of this research has been to create one or more cognitively-inspired computational models, then a subgoal would be that the models must exhibit analogs to these processes. If such processes are present in the computational model, then the models' performance on certain tasks may be characterized, and contrasted where appropriate with human performance on those tasks.

Even so, these acts are available to be performed not only due to some variety of processes, but also because some sufficient representation of the received visual scene which affords them.

Representation of knowledge

The representation of the visual scene received by the agent contains information and knowledge about the world from which the scene is taken. In the preceding section, I described the feature integration theory of Treisman and Gelade (1980) as a means by which humans come to associate visual features and information with objects inferred from the scene (perhaps using the vision theory of Marr (1982)). But as I am dealing in artificial intelligence and computational models, how might this be accomplished in an intelligent agent? The information used by the agent must be more than a data structure containing the scene: it should be organized into a knowledge representation.

While use of the term "representation" is quite commonplace in the artificial intelligence literature, what is a knowledge representation? In their paper, Davis et al. (1993) note that knowledge representations play five distinct, critical roles:

- as a surrogate;
- as a set of ontological commitments;
- as a fragmentary theory of reasoning;
- as a medium for efficient computation; and
- as a medium of expression.

Each of these aspects matters when regarding visual reasoning. The fidelity of the correspondence between the representation as surrogate and the received visual scene of the world affects and informs the possible levels of abstraction. The ontological commitment of what within the received signal to represent (and what to leave out) contribute to the constraints the knowledge representation may impose upon reasoning. The fragmentary reasoning that a knowledge representation affords stems from what inferencing it allows, and how that set of allowed inferences may be constrained. The guidance a knowledge representation gives for computation arises from its role as an organizational mechanism for the corresponding received information, and reflects upon the adequacy with which that information is captured. The utility of the knowledge representation for communicating information directly affects the agent's ability to mix new data with old into newer data, and provides the way in which comparison arises.

In the subsequent two chapters of the dissertation, I develop a particular knowledge representation, and discuss in detail why that representation is indeed a knowledge representation.

Vision, and visual reasoning

It may be tempting to view these remarks, and indeed all of my research, as being focused on vision. While there has been substantial research on the detection of objects and novelty in computer vision (e.g. Markou & Singh, 2003a,b; Viola & Jones, 2001), my efforts concern themselves with visual reasoning, and in particular the role analogy-making may play when reasoning about visual stimuli. This dissertation is about cognitively-inspired computational strategies, and how they arise from the choices made when representing a received visual scene.

Challenges

My research has been in the development of a computational model which while addressing a task of visual analogy exhibits cognitively-inspired processes which can distinguish between the familiar and the novel, and which can shift between levels of abstraction automatically. These processes are sanctioned by some appropriate representation of the received visual scene which affords re-representation to varying levels of abstraction and offered features which may be used for memorization, recall, and comparison, as required by the visual analogy tasks. The act of characterizing and developing those processes were thereby co-mingled with the act of describing a suitable representation.

To do this, I identified several specific challenges, which I now detail.

The Challenge of Complexity in Representation

Intuitively, the whole visual world is profoundly messy, and the visual signal received from it is complex. A suitable representation would need to be able to capture the inherit complexity of the received world. However, to demonstrate that the attempt to characterize and capture every conceivable aspect of the world with sufficient complexity would seem intractable.

This challenge contributed to this research in two important ways. It constrained the work to a subset of the world, and thus focused upon visual scenes that are relatively simple and largely geometric in aspect. However, to avoid loss of generality, this simultaneously forces the consideration of a more universal visual representation, a substrate upon which a received visual scene may be built. It was in facing this challenge that I turned to fractals.

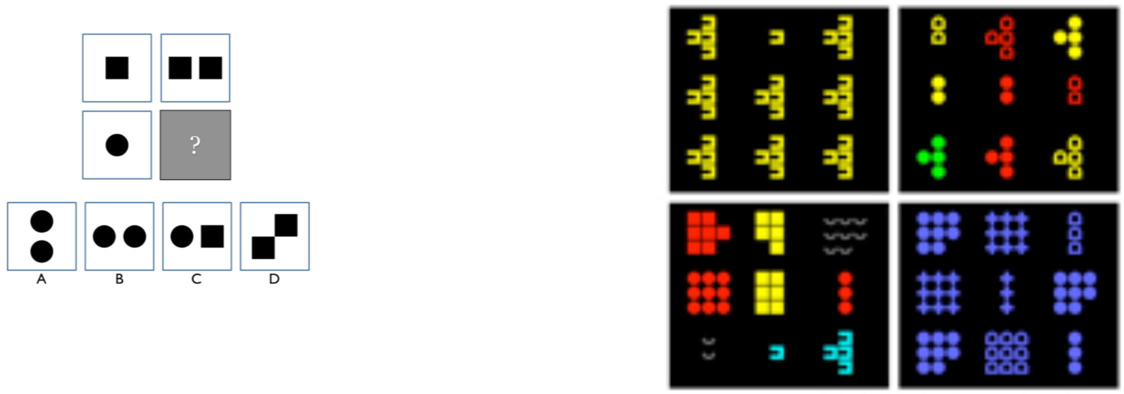


Figure 1.4. Problems of Similarity and Novelty.

The Challenge of Domains

This would be a corollary to the challenge of representation. The complexity of the world does offer up quite the variety of problems and puzzles. While minds might capably perform many, many tasks, this research was focused on the exploration of novelty and abstraction tasks, in the context of visual reasoning specifically. Thus, I restricted consideration of problems to receive from the world to those domains in which novelty or similarity may be determined via visual input alone. Figure 1.4 illustrates two such example problems.

In particular, I chose to restrict the problem domain over which the computational model would operate to tasks of visual analogy. There exists significant prior research into visual analogy (e.g. Goldschmidt, 2001; Davies & Goel, 2001; Ferguson, 1994; Forbus et al., 2008; Hofstadter, 2008). Some of the problem domains addressed by that research have been in the area of computational psychometrics (Bringsjord & Schimanski, 2003; Lovett et al., 2007, 2010; Lovett et al., 2008; Kunda et al., 2010, 2011, 2012, 2013).

It has proven somewhat daunting to create models and write code which will be compared against others' code, and it has certainly been true that one's model and code must achieve a certain measure of correctness on those psychometric tests in order to be taken seriously in literature reviews. However, the selection of problems from psychometrics had a distinct advantage over other domains, in that there exists a general breadth and availability of human performance data on those tests.

The Challenge of Visual Reasoning, Itself

In artificial intelligence, models are built of cognition and computational creativity, and those models are subjected to various tests. Often, these tests are themselves artificial, contrived to limit the model's domain to a carefully composed world (classically, the Blocks world (Winograd, 1973; Bobrow & Winograd, 1977)). Yet, critics of AI charge that the composition of the problem domain itself is too carefully constrained, and that the resulting model clearly should work, for it, and the world upon which it acts, are joined one to another, representationally intertwined (Reeke & Edelman, 1988; Brooks, 1991).

There may be many different ways in which a problem may be represented. However, a chosen representation expressly determines the nature of the reasoning which may operate upon the representation. The selection of representation then must expressly afford and sanction the kinds of visual reasoning the research wished to explore. Thus, the selection of representation was restricted to those which both afforded reasoning about novelty and similarity, and supported shifting levels of abstraction.

The Challenge of Correspondence

The current theories of visual and analogical reasoning depend upon a significant theoretic leap: that the received world is transformed from a series of received percepts into some symbolic representation (e.g. Marr, 1982; Kokinov & Petrov, 2001; Holyoak & Hummel, 2001; Barsalou, 2008). The challenge is that this transduction of perception into symbolism readily may be viewed as reducing correspondence with the world (that is, with reality) (Markman, 1999; Davis et al., 2003). Reducing correspondence with reality affects the correspondence in level of abstraction which might be afforded by the symbolic representation.

Thus, for the purposes of this research, a suitable representation must maintain as strong as practical a correspondence to the received percepts. The need for such strong grounding also was a determining factor in my choice to focus on fractals, the argument for which I develop in the next two chapters..

The Challenge of Models

As mentioned prior, there exist several kinds of visual analogy problems, and this research addresses certain of these. These problems share many common aspects, but they have very specific differences as well. Perhaps it may be assumed that each of these problems quite naturally might lead to its own computational model.

Suppose, instead, that a common representation may be shared amongst those models; would that representation provide an account for their commonality? If so, then one may find that though there can be differences in model, there may exist a single cognitively-inspired computational architecture upon which those models are founded (Johnson-Laird, 1983; Laird et al., 1987; Tversky, 1993).

Finding first these computational models, and then explicating an underlying overall model, has been a goal of this research. In the chapters of this dissertation concerning the various problem domains, I describe the algorithms that constitute the computational model addressing the problem, but I also describe the lineage between the algorithms, and through those connections establish a common model, rooted in the chosen representation.

The Challenge of Judging a Model

According to Cassimatis et al. (2008), computational modeling is a particularly important part of understanding higher-order cognition, for two reasons. First, having a precise model clarifies notions such as representation and concept. Furthermore, being instantiated into a computational model makes the possibility of intelligence arising from natural phenomena more plausible (Cassimatis et al., 2008).

Such models are judged by their degree of ability, empirical coverage and their parsimony (Cassimatis et al., 2008). Judging the ability of a computational model does not mean that there is a direct mapping implied between the performance of the computational model and that of a human: Cassimatis et al. (2008) point to Logic Theorist (Newell et al., 1958) as proof that what mattered was the demonstration that some mechanism could explain some kinds of problem solving. Judging parsimony is straightforward: merely note the number of computational methods needed to address the problem. Judging empirical coverage, in contrast, is complicated.

Ordinarily, empirical coverage for computational models has meant that human performance levels are achieved, both in the time taken to perform a task, and in the number and kind of errors made during a task. However, to judge a computational model's coverage based on time performance must be reconsidered, for at least two reasons. Firstly, each year machines and devices grow faster and faster, and storage more abundant (Schaller, 1997): at some point, the task that satisfactorily covers human performance will be performed much quicker by machine. Thus, the time performance metric, as a standard for empirical coverage, diminishes. Secondly, the algorithms one now designs are generally executed in a serial fashion, with strict data flows. Human brains, in contrast, don't quite seem to follow either aspect, being inherently (and massively) parallel, and with bidirectional information flowing (Ullman, 1995).

For these two reasons, I suggest that judgment be passed upon the research's computational models' empirical coverage in two ways: by its error patterns vis-a-vis human error patterns where available, and by its demonstrated suitability across multiple problem domains.

Limitations

My dissertation's research question is:

How might a cognitively-inspired computational model receive a visual scene and, in the service of some visual analogical task, notice novelty at an appropriate level of abstraction?

The challenges just enumerated offered ways with which to constrain the question's exploration. In light of those constraints, I additionally and deliberately limited the scope of this work in two important ways, namely:

- the work makes a strong commitment to a particular kind of representation; and
- I focused on developing cognitively-inspired computational models for four specific, interrelated problem domains.

Limitation 1: Commitment to a representation

The representation chosen is the fractal representation, a novel visual representation which I developed over the course of performing this research. Indeed, this representation is perhaps the key contribution of my research.

The fractal representation arises from the fractal encoding of visual input. Fractal encoding itself is an encoding of both spatial and photometric relationships which captures the nuances of textures present within a received image. Fractal representations capture the similarity between visual images, even if the images are the same. A thorough discussion of fractal encoding and the development of the fractal representation are found in the subsequent two chapters.

Limitation 2: Commitment to specific domains

In service to the development of this research, I discovered, developed and tested cognitively-inspired computational models based on fractal representations across four specific problem domains, two of visual similarity and two of visual novelty.

For the visual similarity domain, I chose the Ravens Progressive Matrices tests (Raven et al., 2003) and the Miller Analogies Test (Meagher, 2006; Pearson, 2011), used by Evans in his seminal early work on analogy (Evans, 1964). The Ravens tests offer a combined set of 204 well-documented, human-tested visual analogy problems, distributed across four distinct test sets. The Miller Analogies Test offers 20 such problems. The Ravens test and the Miller Analogies test have a similar structure: given a matrix of figures in which one figure is missing, choose from a set of candidate answer figures which one best completes the matrix.

For the visual novelty domain, I choose two particular sets of problems. The first chosen, the Odd One Out test, developed by Adam Hampshire and colleagues at Cambridge Brain Sciences (Owen et al., 2010), consists of almost 3,000 3x3 matrix reasoning problems organized in 20 levels of difficulty, in which the task is to decide which of the nine abstract figures in the matrix does not belong (the so-called "Odd One Out"). For the second set, I chose the Dehaene test of core geometry (Dehaene et al., 2006), consisting of 45 problems designed to measure whether an individual has a notion of certain principles of geometry. Both the Odd One Out test and the Dehaene core geometry test have a similar structure: given a matrix of figures, decide which one does not belong.

My intention was that by considering problems of similarity (Ravens and Millers) independently from problems of novelty (Odd One Out and Dehaene), distinct models would emerge, one for similarity and one for novelty. From these models, the intention was to extract those domain-generic techniques to form the basis of a cognitively-inspired computational model, one in which noting novelty and adjusting levels of abstraction are fundamental and strategic acts, afforded expressly by the fractal representation.

I point out that these problem domains are static, 2D worlds. If this research were concerning itself with vision in the general sense, one would have to choose additional dynamic domains which would offer the opportunity to address the challenges of occlusion, motion, noise, and the like. Although I believe this work may hold promise in those areas, as noted above this research is not about vision: it is about visual reasoning, and the role analogy-making and representation play in it. Nonetheless, I did explore the potential connection with vision and other visual perception, by performing minor experimentation in using fractal representations and similarity as latent support for flocking behaviors in agents, and in modeling perceptual instability when considering the Necker cube.

These domains, the developed computational models, and the results of all experiments are presented in detail in the subsequent chapters devoted to each.

Thesis & Hypotheses

With these challenges, limitations and intentions in mind, I make the following sufficient, expressive thesis statement and collection of hypotheses.

The Thesis Statement

My dissertation concerns itself with this thesis statement:

Reasoning using fractal representations is a novel, feasible and useful computational technique for solving certain problems of visual similarity and novelty.

I developed three primary hypotheses from this thesis statement, and the balance of the dissertation provides a detailed account of my research to confirm them. The three hypotheses are:

- that using the fractal representation, a robust cognitively-inspired computational strategy may be determined which automatically adjusts the representation to an appropriate level of abstraction;
- that using the fractal representation, a robust cognitively-inspired computational model can be derived for certain classes of problems of visual similarity, such as the Raven's Progressive Matrices tests; and
- that using the fractal representation, a robust cognitively-inspired computational model can be derived for certain classes of problems of visual novelty, such as those in the Odd One Out set.

In addition to these three primary hypotheses, I make a zeroth hypothesis concerning the representation chosen, that the fractal representation is a knowledge representation.

Hypothesis 1

Using the fractal representation, a robust cognitively-inspired computational strategy may be determined which automatically adjusts the representation to an appropriate level of abstraction.

In support of hypothesis 1, I discovered, developed and implemented an original and novel algorithm, the Extended Analogy By Recall (ABR*) algorithm. The ABR* algorithm is based on the premise that analogy begins by being reminded of something, and integrates the return of a measure of similarity with a retrieved analog. Furthermore, the algorithm also provides how the ambiguity or uncertainty with which an answer to a visual analogy problem may be characterized can be attributed to those features naturally arising from fractal representations. I showed that such a characterization can be used concurrent with problem solution, as a mechanism for driving level-of-abstraction refinement.

Beginning with Chapter 2's robust discussion of the fractal representation and continuing through Chapter 4, using as a visual similarity task as a basis, my dissertation presents a complete description of the algorithm, its motivation, and an argument that the reasoning embodied therein may be construed as a computational model of visual abstraction.

Hypothesis 2

Using the fractal representation, a robust cognitively-inspired computational model can be derived for certain classes of problems of visual similarity, such as the Raven's Progressive Matrices tests.

In support of hypothesis 2, I describe herein the problems of the Raven's Progressive Matrices tests, in terms of their individual nature as well as their importance in the realm of human psychometrics. I developed both a visual reasoning strategy and an algorithm which embodies that strategy, based upon and relying solely upon the fractal representation of a Raven's problem, which can, as shown in Chapter 5, solve the problem without intervention. This dissertation clearly illustrates the algorithm as implemented in both pseudo-code and in the Java programming language (available on our research lab's website), and presents the results of the algorithm's execution against the full set of Raven's Progressive Matrices tests.

Similarly, in Chapter 6, the dissertation shows the same for the problems of the Miller Analogies Test, with no modification to the underlying algorithm or representation. The performance of the Fractal Raven and Fractal Miller algorithms compares quite favorably to all prior computational approaches to these problem domains, as I illustrate in the corresponding chapters.

Hypothesis 3

Using the fractal representation, a robust cognitively-inspired computational model can be derived for certain classes of problems of visual novelty, such as those in the Odd One Out set.

In support of hypothesis 3, I first describe, in Chapter 7, the problems of visual oddity, in terms of their individual nature as well as their distinction from visual reasoning as required for problems of the Raven's test. I developed a visual reasoning strategy and an algorithm which embodies that strategy, based upon and relying solely upon the fractal representation of an Odd One Out problem, and demonstrated that the algorithm will solve the problem without intervention. I wrote the algorithm in code and executed that code against a large corpus of Odd One Out problems (approximately 3,000, at varying levels of human difficulty). In Chapter 8, I report the results of the algorithm's performance.

In addition, the visual oddity algorithm developed for the Odd One Out was extended and used to address those problems present in the Dehaene set of core geometry. This dissertation presents the results of those experimental runs as well, in Chapter 9, and compares those results with a prior computational approach to the Dehaene set as well as to human results.

The Zeroth Hypothesis

The fractal representation is a knowledge representation.

In support of this hypothesis, in Chapter 2 I fully motivate, develop and illustrate the fractal representation. Furthermore, I present the manner by which the fractal representation may be extended, from a representation of a single image, to any number of images. In Chapter 3, drawing on the criteria and roles of knowledge representations in general of Markman, Davis, and others, I illustrate precisely how the fractal representation satisfies the various criteria, and thereby is to be regarded as a knowledge representation.

Additional Results

I explored two additional problem domains with the fractal representation, using the core machinery of the Extended Analogy by Recall (ABR*) strategy.

In Chapter 10, I report the details of an exploration of fractal perception in interacting agents. In this chapter, I describe first how computer graphics simulations of flocking agents occurs. I implemented such a simulation, which contained hundreds to thousands of those agents. I then introduce the idea of providing a perceptual processing system to one of those agents, based on fractal representations, and show how the behavior of the agent can remain analogous to the other agents, as a full participant in the flock.

In Chapter 11, I report a computational model of perceptual bistability, using the Necker cube as the subject of study. I provide a review of prior attempts to characterize perception of the Necker cube, and then, through the use of three sets of exemplar images, present the ambiguous Necker cube, represented fractally, to the Extended Analogy by Recall algorithm. In my results, I show that the Necker cube as perceived by my computational strategy remains in a perceptually ambiguous state, regardless of exemplar or level of abstraction. To my knowledge, this is among the first ever computational models of perceptual bistability. Furthermore, these results suggest that my computational model offers potential avenues in which to explore additional cognitive phenomena.

Contributions

My dissertation and the body of research it describes makes two primary, novel, and significant contributions to science.

The first contribution is the Extended Analogy by Recall (ABR*) algorithm, a parsimonious, cognitively-inspired computational model for visual reasoning which automatically adjusts its representations to an appropriate level of abstraction. The subsequent chapters of this dissertation show unmistakably that the strategy contained within the ABR* algorithm is suitable to meet the demands of a variety of visual analogy problems. In addition to this primary contribution, several algorithms, which address reasoning specifically in visual similarity and visual oddity tasks, as well as algorithms which afford or mimic aspects of visual perception, are contributions in their own right.

The second contribution is the fractal representation itself, a new and novel knowledge representation that will open the door for analogy researchers, cognitive scientists, and computer scientists to explore the role self-similarity and perceptual complexity play in analogy making.

The concluding chapter of this dissertation expands upon a number of potential future research directions suggested by these contributions.

A Guide to this Dissertation

This dissertation is divided into five sections.

The first of these sections is introductory in nature, and includes the present chapter and a chapter on knowledge representation and the fractal representation. The details of exactly how to transform an image into the fractal representation are presented in that chapter, and pseudo-code is provided.

The second section concerns itself with visual reasoning, and in particular the visual similarity domain. Chapter 4 develops the overall approach to fractal visual reasoning and introduces the visual similarity algorithm. Later in that chapter, I continue the refinement of visual reasoning, incorporating levels of abstraction and automatic abstraction shifting, and introducing the Extended Analogy By Recall (ABR*) algorithm.

In the third section, the visual similarity domain is explored by example. Chapter 5 presents the development of the Fractal Ravens algorithm and shows, by way of extensive example, how it may be used to solve problems of the Ravens Progressive Matrices tests. The particular results of the algorithm upon the Ravens tests may be found in Chapter 5, along with comparison to human performance data. Chapter 6 concerns the adaption of the Fractal Ravens algorithm into the Fractal Miller algorithm and its use on the Miller's analogy problems.

The fourth section devotes itself to the visual oddity domain. Chapter 7 builds upon the lessons learned from the Fractal Ravens algorithm, and introduces the visual oddity algorithm. The Odd One Out problems suite is discussed at length in Chapter 8, and the results of running the ABR* algorithm against the large corpus of oddity problems. Chapter 9 is an explication of the Dehaene core geometry problems, and presents the results of an experiment in which the ABR* algorithm is made to address them.

The final section summarizes the dissertation and its implications. It begins with two chapters (Chapters 10 and 11) which describe the preliminary experiments into fractal perception and the emergence of perceptual instability when using the model to reason about a classic problem of perceptual gestalt psychology. The final chapters

(Chapters 12 and 13) provide a review of the claims made by this dissertation, the implications of this work for various fields, and a glimpse into future research directions.

CHAPTER 2

FRACTALS AND REPRESENTATION

In this chapter, I discuss the development and construction of the fractal representation, as a powerful means of representing images.

Fractal Encoding and Representations

An image, as held in memory in a computer, is a representation which may occur in a variety of forms. In one case, a vector image, the image might be represented as a proximal sum of a variety of lines, curves, and polyhedral shapes. Vector images are quite well suited for representing diagrams. In another, more common example, the image might be represented in bitmap fashion, a rectilinear array of pixels (photometric values) of a specific width and height. Bitmapped images are typically used as methods for storing so-called “natural images.” In either case, a coordinate system typically is inferred to ascribe the position and orientation of various spatial elements, be they pixels or polygons.

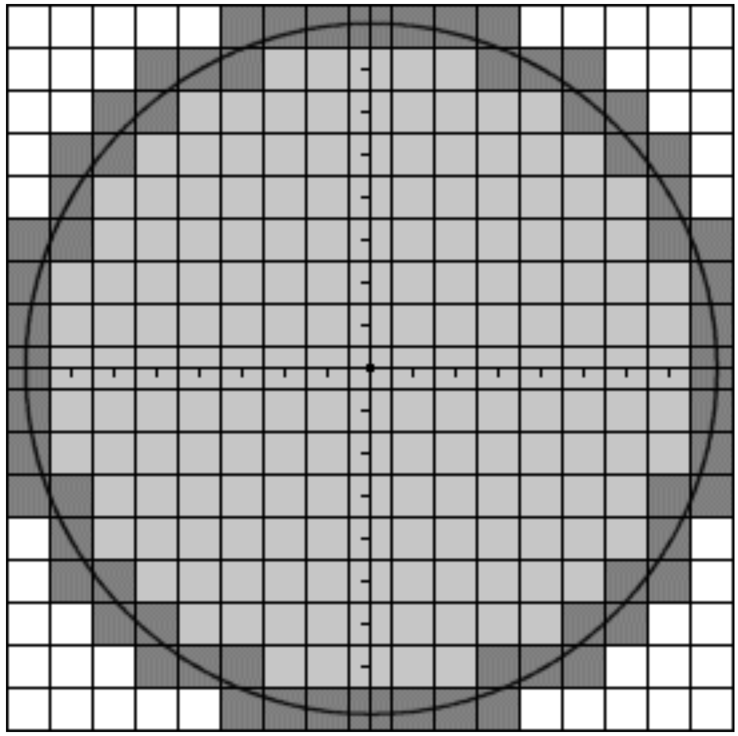


Figure 2.1. A Circle, As Pixels.

The challenge of representing an image, in any fashion, stems from this: to what end is the representation intended? As shown in the previous section, a representation entails a set of possible inferences, and implicates a surrogate standing. An image representation is arrived at from some putative input. We receive the world, and we represent it.

Fractals

Benoit Mandelbrot coined the term "fractal" from the Latin adjective *fractus* and its corresponding verb (*frangere*, "to break" into irregular fragments), in response to his observation that shapes previously referred to as "grainy, hydralike, in between, pimply, pocky, ramified, seaweedy, strange, tangled, tortuous, wiggly, wispy, wrinkled, and the like" could be described by a set of compact, rigorous rules for their production (Mandelbrot, 1982).

The computer graphics community has generated fractal imagery, similar to this figure, for several decades. Indeed, there are several different kinds of fractals described within the literature of computer graphics, physics, and mathematics. Here are a few examples.

Figure 2.2. A Fractal Fern, Constructed From Other Ferns.

Iterated Function Systems

Iterated function systems (IFSs) were devised by Barnsley and his colleagues (Barnsley & Demko, 1985) as a means of generating a broad class of fractals, using a set of affine maps and an associated set of probabilities. Each such IFS resolves to a single attractor set. In the early 1980s, much effort was focused on the generation of naturally occurring, complex phenomena, such as clouds, plants, and landscapes. Demko et al. proposed the use of iterated function systems as one such method for creating computer graphic models of these phenomena (Demko et al., 1985). The relatively small set of affine maps and overall compact nature of IFSs was demonstrated by Demko et al. successfully. The iconic fractal fern shown in figure 2.2 is due to a three-map IFS discovered by Barnsley. Iterated function systems have been used to model single-valued discrete-time sequences (Mazel & Hayes, 1992), neural networks (Stark, 1991), and image compression (Barnsley & Sloan, 1990). It is from this later work, as image compression, that my own research and development of fractal representations stems.

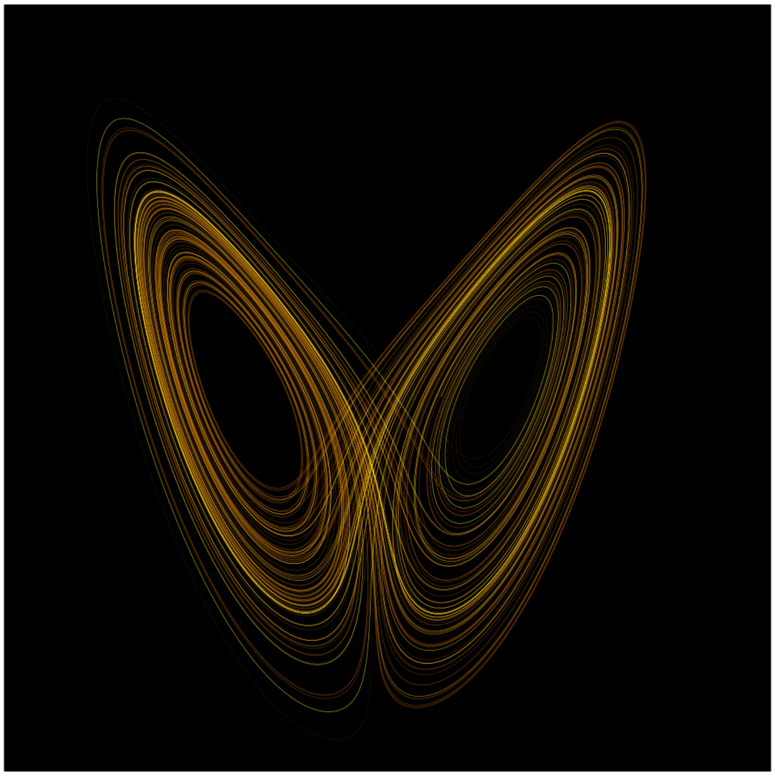


Figure 2.3. Strange Attractors

Strange Attractors

Studies of turbulence in fluid mechanics and nonlinear physics gave rise to a more mathematical class of fractals known as strange attractors (Grassberger & Proccaccia, 1983; Eckmann & Ruelle, 1985). In a physical model, the whole system is represented by a number of modes - independent oscillators of variables, or states. While each mode can be thought of as periodic, the whole system is quasi-periodic (a superposition of the modes), and a system can be seen as progressively more turbulent as the number of modes increases (Eckmann & Ruelle, 1985). An attractor is a mathematical description of the stable oscillation of the dynamic system as transient behaviors decrease. However, there are systems in which this behavior itself is unstable. Such systems are deemed chaotic by Eckmann & Ruelle (1985) and thereby possess a strange attractor. A companion way to consider a strange attractor is to think of the attractor itself as a particular configuration of a system (set of states) and a probability that the system would be in that particular configuration (Halsey et al., 1986). In this way, one can make an analogy that the set of states and probabilities view of strange attractors is similar to the iterated function system notion of set of affine transformations and probabilities.

Figure 2.4. Lindenmayer Systems

Lindenmayer Systems

As early as 1968, Lindenmayer developed mathematical models of cellular interaction and growth (Lindenmayer, 1968). Lindenmayer (Lindenmayer et al., 1990) fully realized his system of describing plant growth as a technique of originating with a string of symbols (an initial condition) and a set of rules for transforming a substring of symbols into a different set of symbols. Each of these rule might be interpreted as an instruction for generating part of a plant (grow longer, branch left, fork at a certain angle, etc.). Probabilities could be associated with the advent of each rule.

As computer graphics techniques improved through the 1980s, Smith (1985) turned to Lindenmayer’s descriptions of string-rewriting rules as a method for generating realistic renditions of plants. These fractal plants were determined both by their initial conditions, and by the probabilistic choice of which of several rewriting rules would be used during construction. The advent of procedural modeling in computer graphics saw Lindenmayer systems applied to a variety of models, including the modeling of cities (Parish & Muller, 2001).

Again, it may be seen that the notion of a set of rules and an associated set of probabilities is very much akin to that of an iterated function system.

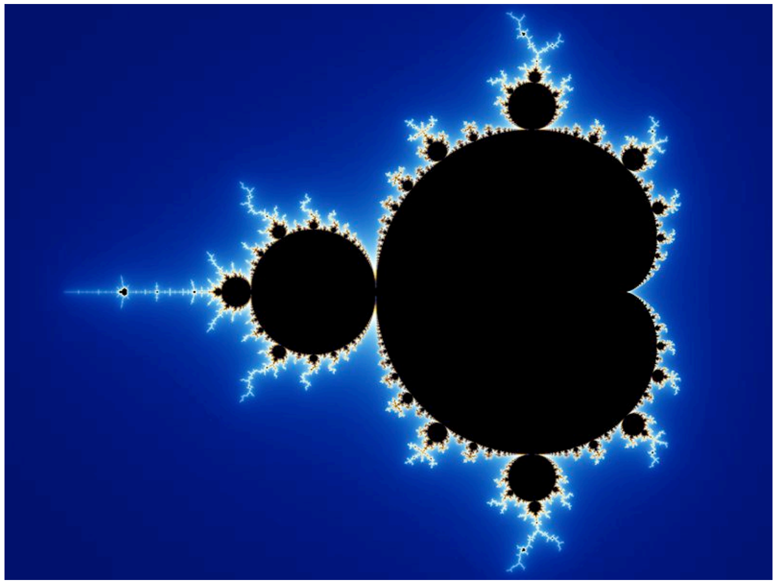


Figure 2.5. Escape-time Systems

Escape-Time Systems

Renderings of Julia sets and Mandelbrot sets are the most commonly seen images associated with the word "fractal." Both of these sets, however, are examples of escape-time systems. In an escape-time system, for each point in a set there exists a recurrence relationship. That is, when one arrives a particular point in the system, there is a function over that point in the set which maps the point to another point in the set. The colorization of the renderings of a Julia or Mandelbrot set can be thought of as the distance between the point and its subsequent mapping.

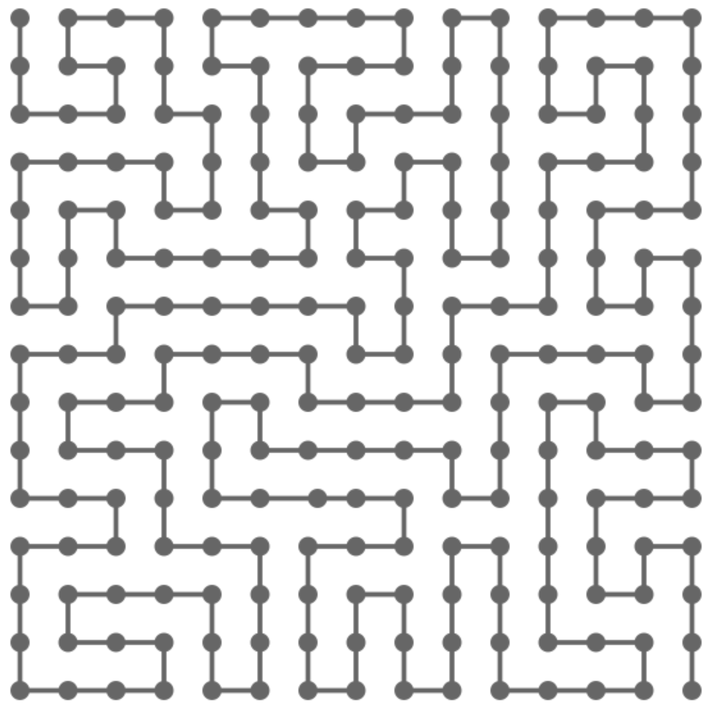


Figure 2.6. Random Fractals

Random Fractals

Space-filling curves and surfaces such as Hilbert curves, Koch snowflakes and Sierpinski gaskets as well-known, and have broad application (e.g. Baliarda et al., 2000 for a description of using Koch curves for compact antenna design). In general, these surfaces and curves are formed through the consistent and repeated application of a specific rule, or set of rules: for Koch snowflakes, replace any line segment with four segments, each one third the length of the original, with the middle third of the original segment replaced by two segments joined as if to form an equilateral triangle with the original middle third.

However, the mathematical regularity of such curves and surfaces can be perturbed by deciding, based on some stochastic method, when and where to apply the rules. Such a process, while chaotic, can generate curves and surfaces with the same fractal nature and the same expressive natural rendering potential (Krapivsky & Ben-Naim, 1994; Mandelbrot, 1975; Schenider & Westermann, 2001). These random fractals begin with some initial condition (to which they are sensitive), and then extend the general notion of set-of-rules plus probabilities for application to include the idea that the probabilities themselves may vary. Put differently, the random fractal has a form which contains the transformation rules and a range of putative probabilities for each, rather than a single value.

<u>Fractal imagery</u>

While the various formulas for generating fractal imagery is quite well-known, many images of real-world artifacts appear to have "fractal" properties. Indeed, the quest to render apparently real-world artifacts propelled the discovery and use of fractal descriptions and techniques in computer graphics as noted above. If these images are "fractal" in some sense, then what formula (to be more specific, what representation) may underlie these images? We must now consider a fundamental question: what, precisely, is fractal?

Fractals in the Real World

The mathematical derivation of fractal image representation expressly depends upon the notion of real world images, i.e. images that are two dimensional and continuous (Barnsley & Hurd, 1992). Both of these assumptions are important. That an image is two dimensional means that there is an ability to assign a coordinate system to the image, and that the photometric elements, the pixels, within that image have a spatial relationship to one another (that there is a distance metric upon the space). That an image is continuous implies that no matter how closely one might choose to examine the image, there still will remain finer and finer gradations of the pixels. In a sense, the continuity of the image suggests that the selection of an image's resolution (the ability to resolve or describe a single pixel) is under the control of the observer. In this assumption, a pixel gains the descriptive quality of a photometric region.

Figure 2.7. Images with Fractal Properties.

Real world imagery, in the definition above, encompasses not only that which occurs in the natural world, but all imagery. Natural and artificial scenes, all diagrams and schemata, every image which arises as a result of light being reflected by or transmitted from any surface and subsequently falling upon the photoreceptors and made available to the human visual system is a real world image. Images generated internally or those arising from some act of visual imagination or via some other means (specifically, those images whose arrival does not encompass perception and the enactment of the early visual system, the lensing system, and especially, the striate and pre-striate cortex) are excluded from the definition of real world imagery.

Figure 2.8. A Field of Sunflowers, Showing Repetition and Similarity at Scale.

A key observation by Barnsley and Hurd (1992) is that all naturally occurring images perceived appear to have similar, repeating patterns. Another observation is that no matter how closely you examine the real world, you find instances of similar structures and repeating patterns. The twin ideas, of repeating patterns and of repetition at differing scales (or resolution), combine to provide the basis for labeling such images as “fractal.” Importantly, the repetitive nature of these images persists at all observable scales, down to the resolving power of the observer.

Figure 2.9. Broccoli, Illustrating Similarity and Repetition.

These powerful observations suggest that it is possible to describe the real world in terms not of traditional graphical elements, but of observed similarity and repetition alone. This is the crucial idea upon which the fractal representation is formulated.

The Mathematical Basis for Fractals as Operations

The mathematical derivation of fractal representation as an operation over images expressly depends upon the notion of real world images, i.e. images that are two dimensional and continuous (Barnsley & Hurd, 1992). Every image received by the human visual system may be construed as meeting this requirement, with the proviso that the notion of continuity has a resolution limit, and that limit plays a significant role in visual abstraction, as shall be discussed later in this dissertation.

Collage Theorem

Computationally, the determination of the fractal representation of an image can be performed through the use of the fractal encoding algorithm. The collage theorem (Barnsley & Hurd, 1992) at the heart of the algorithm can be stated concisely:

For any particular real world image, there exists a finite set of affine transformations which, if applied repeatedly and indefinitely to any other real world image, will result in the convergence of the latter into the former.

It is important to note that the collage theorem is describing a set of transformations which are derived by mapping an image into another. In other words, fractal encoding determines an iterated function system which is applied repeated to some source image, with the result that the encoded image emerges.

The Development of the Fractal Representation

Let us suppose F() is a fractal encoding of image B. Then, since I have designated fractal encoding as an iterative function system, the successive application of F() onto its prior output will converge upon the image B. Thus, given any other image A:

$$\mathbf{F(A) = A_1, F(A_1) = A_2, F(A_2) = A_3 \ldots \text{ and so on, until } F(A_\infty) \doteq B}$$

According to the Collage Theorem, F() is itself a finite set of affine transformations T which describe how to modify portions of an image such that convergence is assured. Therefore,

$$\mathbf{F() \equiv T \equiv \{ T_1, T_2, T_3, \ldots , T_n \}}$$

Each of the constituent affine transformation may affect some or all of the given image, but it is the unordered union of their actions which comprises the resultant image. Thus:

$$\mathbf{F(A) = T(A) = \cup\, T_i(A), 1 \le i \le n}$$

Dependencies

There are several, interrelated dependencies implied in the Collage theorem. These are specificity, partitioning, and search. I shall describe each in turn now, in the context of the theorem, and later, in the context of the algorithm and the subsequent representation.

<u>Dependency upon the specificity of the source and the destination.</u>

The first dependency is that such a fractal encoding is dependent not only upon the destination image, but also upon the source image, from which the set of affine transformations T is discovered. However, once the fractal encoding has been determined, the application of that encoding to any source image will result in the target image. This dependency is to suggest that, a priori any application of the encoding, a particular fractal encoding is determined uniquely by a particular source image.

<u>Dependency upon the partitioning of the destination image.</u>

The cardinality of set of transformations is determined exactly and solely by the partitioning scheme chosen for the image being encoded. It is presumed that the image being encoded admits to being partitioned in some manner, however.

Images may be partitioned using a variety of methods. In computer vision, one typically seeks to segment an image into regions or shapes (Ray & Turi, 1999; Zhu & Yuille, 1996). Another segmentation scheme would seek to segregate an image into two segments, a foreground and a background (Kim, et. al., 2005). Other partitionings of images may be regular, such as the division of computer-based images into pixels, at some resolution. It must be noted that the choice of partitioning scheme affects the computational complexity of enacting the partitioning.

The Collage theorem imposes no constraint upon the choice of partitioning save one, and that is that the union of all partitions wholly cover the image to be encoded. Topologically speaking, the image B is treated as a set, and the partitioning P() of that image into a finite collection of subsets is a cover of that set if:

$$\mathbf{P(B) = \{ b_1, b_2, b_3, \dots b_n \}}$$

$$\mathbf{B \subseteq \cup\, b_i,\ 1 \leq i \leq n}$$

Dependency upon the search of the source image.

The essential implied step of the Collage theorem is that there is a match for each subimage of the destination, as determined by the partitioning, to be sought within the source image. Through this searching process, the affine transformation for that subimage is obtained. However, the quality and character of the match, as well as the computational complexity of the algorithm, depends upon the constraints selected for comparing the destination subimage with some portion of the source.

Fractal Encoding Algorithm

Given a target image B and a source image A, the fractal encoding algorithm seeks to discover this particular set of transformations T.

First, choose a partitioning scheme P to systematically divide the destination image B into a set of images, such that $B \subseteq \{b_1, b_2, b_3, \ldots b_n\}$.

For each image b_i:

- Search the source image A for an equivalent image fragment a_i such that an affine transformation of a_i will likely result in b_i.
- Collect all such transforms into a set of candidates C.
- Select from the set C that transform which most minimally achieves its work, according to some predetermined metric.
- Let T_i be the representation of the chosen transformation associated with b_i.

The set $T = \{T_1, T_2, T_3, \ldots\}$ is the fractal encoding of the image B.

Algorithm 2.1. Fractal Encoding of B in terms of A

As can be seen in Algorithm 2.1, the fractal encoding of image B in terms of image A consists of two phases: partitioning and searching. I shall now discuss each phase.

Partitioning

The target image B is first partitioned into a set of other images. As one is dealing with computer images, one may safely assume that the image B will have some finite resolution, and thus a limit as to the smallest achievable partition. Practically, this smallest resolvable unit is a unitary pixel, which would denote both a spatial location and a photometric value.

As noted above, the Collage theorem places a topological constraint on the partitioning scheme, and requires that it form a topological cover over the image B. Such a constraint admits a wide variety of methods for partitioning the image, but many of these partitionings may prove computationally expensive.

In the interest of reducing computational complexity, one may impose two additional constraints: each subimage in the partition must be simply connected to at least one other subimage, and the union of all of the subimages must be exactly equivalent to the image B. Stated mathematically:

$$\mathbf{P(B) = \{ b_1, b_2, b_3, \ldots b_n \}\ \text{is a valid partitioning iff:}}$$
$$\mathbf{\forall\, i\ \exists\, j \neq i : \text{simplyconnected}(\, b_i \cup b_j\,),\ \text{and}}$$
$$\mathbf{B \equiv \cup\, b_i\, ,\ 1 \leq i \leq n}$$

where

$$\mathbf{\text{simplyconnected}(\, X\,) \rightarrow \nexists\, x,y \subset X : x \cap y = \varnothing}$$

One computationally inexpensive way to achieve such a constrained partitioning is to impose upon the image B a uniform, rectilinear grid, and select the subimages based upon some chosen grid size, as expressed in units of pixels.

Thus, a stronger specification of the fractal encoding T may be thought of as a function of three variables, the source image A, the target image B, and the partitioning scheme P:

$$\mathbf{T(\, A, B, P\,) = \{ T_1, T_2, T_3, \ldots , T_n \}}$$

where the cardinality of the resulting set is determined solely by the partitioning P. That is, each subimage bi that P extracts from B will be represented by exactly one element of the set T.

Partitioning and Level of Detail

Choosing a partitioning determines the level of detail at which an image is encoded. Thus, the coarsest level of detail possible for an image is the partitioning into a single image (the whole image). The finest level of detail achievable is that set of images wherein each image is but a single pixel.

Figure 2.10. Levels of Partitioning.

The choosing of a grid size, and of a partitioning in general, may be interpreted as an indication of the level of detail at which an image may be encoded. Figure 2.10 illustrates the effect of partitioning an image into a variety of levels of detail, using a regular rectangular grid.

The ability to express level of detail as an artifact of partitioning, whether by controlling grid size, by altering the consistency of partition size, or by modification of the shape and nature of the underlying regions and their spatial arrangement (i.e. hexagonal versus rectilinear scaffolding, or polar versus Cartesian coordinates) is an important aspect of the encoding, and a key feature entailed by the fractal representation.

Searching

The partitioning scheme P extracts a set of images b_i from the target image B. The next step of the algorithm is to perform a systematic examination of the source image A for fragments of A which can be said to best match a particular image b_i. The method by which the search is conducted may be varied, as can the meaning of what is said to be a "best match."

Global and Local Coordinates

An image b_i extracted by the partitioning scheme can be considered as a region containing a number of pixels which are addressable in some fashion. The addressability of these pixels may be viewed as a local coordinate system imposed upon the region. Additionally, the region described by the image b_i has a location and orientation within the image B, strictly determined by the partitioning scheme. Thus, the image b_i may be considered as an ordered set of pixels, having both a local (intrinsic) coordinate system and extent, and a position and orientation within a global (within image B) coordinate system. Figure 2.11 illustrates the local and global coordinate systems.

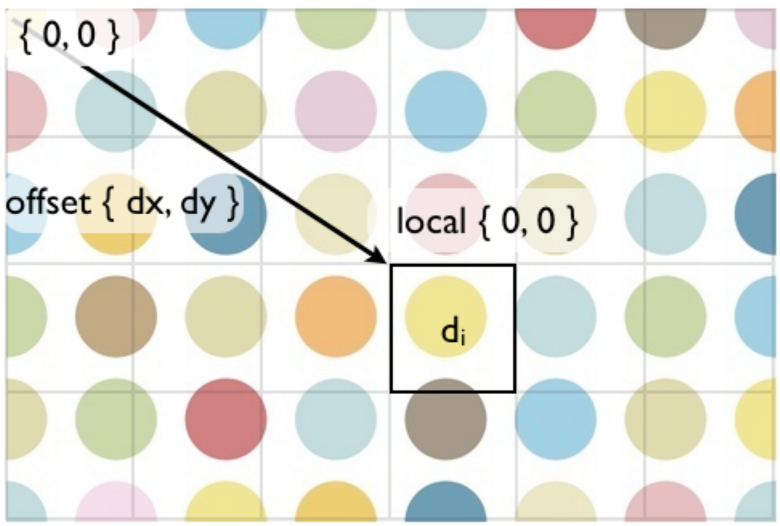


Figure 2.11. Global and Local Coordinates.

However, the same partitioning scheme necessarily does not need to be applied to the source image. The entire source image A may be examined in any manner for a fragment that most closely matches b_i.

Discovering the "Best Match"

The source image A is examined to determine which fragment of it, which I shall label a_k, can be said to "best match" the sought-for image b_i from the target image B. That is, the correspondence between a_k and b_i can be said to be "best" if it is the minimum value of the following function:

$$\textbf{Correspondence}(a_k, b_i) = \textbf{PhotometricCorrespondence}(\textbf{Transform}(a_k, t), b_i)$$

$$\forall\, a_k \subset A,\ t \in \textbf{AdmissibleTransformations}$$

where AdmissibleTransformations is a finite set of spatial transformations applied by the operator Transform() to the pixel values contained within a_k, and PhotometricCorrespondence() is a pixel comparison operation.

Photometric Correspondence

The photometric correspondence between the fragment a_k from the source image A and b_i from the destination image B is calculated to be the difference between the photometric values found in those fragments under a given alignment of their pixels. I wish to propose a metric to ensure that this difference would be 0 if the two fragments were identical photometrically. Such an algorithm to calculate the photometric correspondence is given by Algorithm 2.2:

Let $C \leftarrow 0$.
For each pixel $x \in b_i$ and corresponding pixel $y \in a_k$:
  $C \leftarrow C + (\text{Photometric}(x) - \text{Photometric}(y))^2$
The value C is then the photometric correspondence between a_k and b_i.

Algorithm 2.2. *Photometric Correspondence*

The corresponding pixel in a_k is determined by imposing the same local coordinate system used in b_i upon a_k.

The Photometric value of a pixel used in this calculation may vary according to the nature of the image itself. For example, if the image is in full color, the photometric value may be a triplet of actual values; if the image is monochromatic, then the photometric value will be single valued. Since it is desired to calculate a photometric correspondence which is single-valued, a mapping from multivariate photometry to a single value is typically employed. This can be seen, globally, as mapping from one color space into another. For example, to reconcile traditional computer graphics images given in triplets of red, green, and blue values into single grayscale values, a formula such as this may be used, which seeks to equate the colorimetric luminance of the RGB image to a corresponding grayscale rendition (McGreggor, et al. 1999):

$$\textbf{Photometric}(\langle R,G,B\rangle) = 0.3\,R + 0.59\,G + 0.11\,B$$

Careful consideration of the underlying photometric nature of the image being encoded therefore must be given, but only at this particular moment in the overarching algorithm for encoding. The choice of the Photometric() function determines the interrelationship of the image's colorimetry and its constituent importance to the matching function.

Affine Transformations

The fractal encoding algorithm seeks to find the best matching fragment in a source image which corresponds to a given image partitioned from the target image. As shown above, this matching is achieved by calculating the photometric correspondence function between two fragments, while considering all admissible transformations of the fragment from the source. The set of admissible transformations is a subset of affine transformations known as similitude transformations.

An affine transformation, in two dimensions, may be considered to be of the form:

$$W(x,y) = (ax + by + e,\ cx + dy + f)$$

where a, b, c, d, e, and f are all real numbers. This equation, which maps one point in a two-dimensional plane into another point in a two-dimensional plane, may be rewritten into matrix form like so:

$$W(\langle x,y\rangle) = \begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}$$

In this way it can be seen that an affine transformation is a combination of a linear transformation followed by a translation.

Not all affine transformations are admissible for the fractal encoding transform, however. In particular, those which are admissible must be invertible (Barnsley & Hurd, 1992). Intuitively, this means that each point in space can be associated with exactly and only one other point in space. Mathematically, this means that the inverse has this form:

$$\mathbf{W^{-1}(x,y) = (dx - by - de + bf, -ex + ay + ce - af) / (ad - bc)}$$

and the denominator must not be equal to zero to satisfy invertibility.

Similitude Transformations

An important group of affine transformations are those which are called similitudes. A similitude transformation may be expressed in one of these two forms:

$$W(\langle x,y \rangle) = \begin{bmatrix} r\cos\theta & -r\sin\theta \\ r\sin\theta & r\cos\theta \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}$$

$$W(\langle x,y \rangle) = \begin{bmatrix} r\cos\theta & r\sin\theta \\ r\sin\theta & -r\cos\theta \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}$$

Thus, a similitude transformation is a composition of a dilation factor r, an orthonormal transformation (a rotation about the angle Θ where $0 \leq \Theta < 2\pi$), and a translation (e,f). Similitude transformations are invertible except when $r = 0$.

Defining the AdmissibleTransformations set

Given this formulation for similitude transformations, one can imagine having to consider a great many potential rotational angles to find the best match. Indeed, the computational complexity of the encoding would seem a function of the angles under consideration. In practice, I find that only eight of these orthonormal transformations need to be considered, as shown in Figure 2.12.

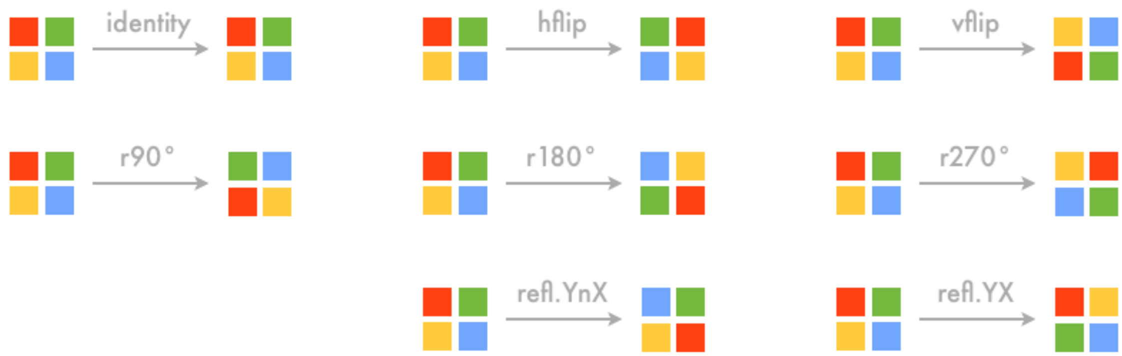


Figure 2.12. The Eight Operations over 2x2 Pixels.

Consider the smallest region of pixels for which orthonormal transformations upon those pixels would result in a visible change. The size of this region is an area two pixels wide by two pixels high. This small region has four lines of symmetry. Taking into account each line of symmetry, and reflecting the pixels in the region about each in turn, there are eight possible outcomes.

My implementation of the fractal encoding algorithm examines each potential correspondence under each of these possible transformations. These form the set of admissible transformations. The transformation from this set which yields the best photometric correspondence is noted by the search algorithm.

Translation arises from searching

The searching process examines each potential fragment in a given source image for correspondence to a particular fragment of the target image. Let us presume that the coordinate systems of the source and the target images may be aligned such that their origins exactly coincide. Then, the relative location of a potential fragment in the source image can be mapped to a location within the target image. This mapping, from the potential fragment's local origin to the particular fragment's local origin, is a translation, and it is this mapping which forms the translation portion of the sought-for similitude transformation.

Dilation and Fractals

Taken together, the orthonormal transformation and the translation provide a sufficient means for describing self-similarity which may exist within an image. However, that self-similarity is not quite sufficient for describing how the similarity may occur at different levels of detail. The dilation factor, r, is used to invoke a contraction of space, whenever $r < 1.0$. The fractal encoding algorithm prescribes that the dilation factor to be used when searching may be conveniently set as $r = 0.5$. In practice, this entails that the source image, as a whole, may be scaled to one-half its original size, and then searched for photometrically corresponding fragments.

Mathematically, choosing $r < 1.0$ ensures that the encoding derived for the entire image, if applied successively and indefinitely to an image, will cause the resulting image to converge upon the desired destination image (Barnsley & Hurd, 1992).

Colorimetric Contraction

As a final step, having located the best photometrically corresponding source fragment, the algorithm determines a rate at which the two regions may be brought into colorimetric harmony. To do this, the average colorimetric description of both regions is calculated, and the distance between the two is multiplied by a dilation. The formula used to calculate the colorimetric contraction is:

$$\mathbf{colorContraction(\ a_k,\ b_i\ ) = 0.75\ *\ (\ colorMean(\ b_i\ ) - colorMean(\ a_k\ )\ )}$$

where the colorMean of a region is the average of all colorimetric information available in that region, taking into account the multivariate nature of the underlying image as previously discussed. The derivation of the colorimetric dilation factor of 0.75 is given by Barnsley and Hurd (1992), and is shown to be correlated to the spatial dilation factor of 0.5.

Exhaustive Searching

The search over the source image A for a matching fragment is exhaustive, in that each possible correspondence a_k is considered regardless of its prior use in other discovered transforms. By allowing for such reuse, the algorithm affords the first Mandelbrot fractal observation, the notion of repetition.

Refining Correspondence

There may be many fragments in the source image which may have identical photometric correspondence to the sought for fragment b_i. This is particularly true when all of the values in the two fragments are identical. To break these potential ties, a further refinement of the correspondence function is necessary.

I compute a simple distance metric upon the images, and give it a weighting. Thus, the correspondence calculated between two fragments becomes:

$$\mathbf{Correspondence}(\, a_k, b_i \,) = w_1\ \mathbf{PhotometricCorrespondence}(\, \mathbf{Transform}(\, a_k, t), b_i \,)$$
$$+\ w_2\ \mathbf{Distance}(\, a_k, b_i \,)$$
$$\forall\ a_k \subset A,\ t \in \mathbf{AdmissibleTransformations}$$

where the weights w_1 and w_2 are chosen such that the calculation of correspondence is dominated by the value of the photometric correspondence. This can be ensured if the following relationship is held:

$$w_2\ \mathbf{maximalDistance} \ll w_1\ \mathbf{minimalJustNoticeablePhotometric}$$

where maximalDistance is the longest possible distance between the origins of bi and any fragment in the corresponding source image, and minimalJustNoticeablePhotometric is the PhotometricCorrespondence which would be calculated if the photometric difference between b_i and any fragment were so small as to be indistinguishable. Practically, I set this value such that this is as small as possible yet not zero, given the color system used in the images. For example, for 8-bit greyscale images where the value 0 represents “black” and the value 255 represents “white,” the minimalJustNoticeablePhotometric would be set to a value of 1.

Fractal Codes

For each image b_i taken from a partitioning of the target image B, the fractal encoding algorithm locates, via exhaustive search over the source image A, a corresponding fragment a_k which the algorithm has deemed to be most minimally distant photometrically under a discovered transformation. The algorithm constructs a description of its discoveries, in a representation called a fractal code. A fractal code consists of six elements, as shown in Table 2.1.

Table 2.1. Elements of a Fractal Code.

Spatial		***Photometric***	
s_x, s_y	Source fragment origin	C	Colorimetric contraction
d_x, d_y	Destination fragment origin	Op	Colorimetric operation
T	Orthonormal transformation		
S	Size/shape of the region		

Note that the dilation factor, for both spatial and photometric properties, is not represented here. This is for efficiency, as these dilations are presumed to be global.

Further efficiencies of expression also may be found by dropping the colorimetric operation (a way of describing how the colorimetric contraction value is to be combined into the region). Since the set of orthonormal transformations the search mechanism uses is finite, I represent the transformation as a referent to that transformation's ordinal membership in the set. The size and shape of the region may be reduced itself, if the partitioning of the image is regular. In my implementation, I use a regular, uniform partitioning, which forms a grid. Thus, the size and shape of the region can be expressed with a single integer, which represents the width and height of the region in pixels.

Arbitrary selection of source

The choice of source image A is arbitrary. Indeed, the target image B may be fractally encoded in terms of itself, by substituting B for A in the above algorithm. Although one might expect that this substitution would result in a trivial encoding (in which all fractal codes correspond to an identity transform), this is not the case, a fractal encoding of B will converge upon B regardless of chosen initial image. For this reason, the size of source fragments considered is taken to be twice the dimensional size of the target image fragment, resulting in a contractive affine transform. Similarly, as shown above, color shifts are made to contract. This contraction, enforced by setting the dilation of spatial transformations at 0.5, provides the second key fractal observation, that similarity and repetition occur at differing scales.

Arbitrary ordinality of encoding

The ordinality of the set of fractal codes which comprise a fractal representation is similarly arbitrary. The partitioning P may be traversed in any order during the matching step of the encoding algorithm. Similarly, once discovered, the individual codes may be applied in any order, so long as all are applied in any particular iteration.

Fractal Representation is Fractal Encoding

The fractal encoding algorithm, while computationally expensive in its exhaustive search, represents the relationship between two images (or between an image and itself) as a much smaller set of fractal codes, an instruction set for reconstituting the relationship, with inherently strong spatial and photometric correspondence. It is through this encoding that the fractal representation of the relationship between those two images is derived. Indeed, the fractal representation is the fractal encoding.

Features from Fractals

The fractal representation of an image is an unordered set of fractal codes, which compactly describe the geometric alteration and colorization of fragments of the source image that will collage to form the target image. While it is tempting to treat contiguous subsets of these fractal codes as features, I note that their derivation does not follow strictly Cartesian notions (e.g. adjacent material in the destination might arise from non-adjacent source material). Accordingly, each of these fractal codes can be considered independently, and candidate fractal features can be constructed from the individual codes themselves, and not from clusters of codes.

Each fractal code yields a small set of features, formed by constructing subsets of its underlying six-tuple. These features are determined in a fashion to encourage both spatial- and photometric-agnosticism, as well as specificity. My algorithm creates features from fractal codes by constructing subsets of each of the six members of the fractal code's tuple.

I further chose to represent each feature as a concatenated string in memory. I form these strings by attaching a character tag to each field in the fractal code and then converting that field into string format prior to concatenation, like so:

$$\mathbf{s_x, s_y \text{ (source fragment origin)} \rightarrow Ss_xs_y \text{ (string representation)}}$$

The choice of the particular tag is arbitrary, but tagging itself is not: tagging is necessary to avoid in-string matching between the different kinds of fields (e.g. an numerical value may appear in multiple fields of a fractal code). Doing so attributes a world grounding to each field, and collectively to the entire fractal code.

Number of Features

As mentioned above, the features are constructed by extracting subsets of each of the six members of the fractal code's tuple. In theory, the number of available subsets would be equivalent to the size of the power set of those six members, or $2^6 = 64$. However, we may generate more that that number of features, by noticing that two of the primary features (the source fragment origin and destination fragment origin) themselves consist of pairs of numbers. Therefore, the available number of features is at least 1024 (2^{10}). Moreover, we may further extract additional features by considering these two pairs under a different coordinate system (polar coordinates, as an example, would yield a distance and an angle, if we take the destination to be the origin, and the source fragment origin as an endpoint for a ray).

In practice, I notice that the photometric operation typically is constant (corresponding to a "copy" operator), and that the source fragment and destination fragment origins may be treated without loss of generality as their constituent portions. Therefore, the number of available features is closer to 128 (2^7). This is further diminished as we consider that we want the order of the features in each subset to be unimportant.

The selection of which features to use, and whether to numerically combine or repurpose them via alternate coordinate systems, may be viewed as an additional control mechanism for algorithms which operate over fractal representations.

Mutuality

The analogical relationship between source and target images may be seen as mutual; that is, the source is to the destination as the destination is to the source. However, the fractal representation is decidedly one-way (e.g. from the source to the destination). To capture the bidirectional, mutual nature of the analogy between source and destination, I now introduce the notion of a mutual fractal representation. Let us label the representation of the fractal transformation from image A to image B as T_{AB}. Correspondingly, let us label the inverse representation as T_{BA}. I shall define the mutual analogical relationship between A and B by the symbol M_{AB}, given by this equation:

$$\mathbf{M_{AB} = T_{AB} \cup T_{BA}}$$

By exploiting the set-theoretic nature of fractal representations T_{AB} and T_{BA} to express M_{AB} as a union, the mutual analogical representation affords the complete expressivity and utility of the fractal representation.

Extended Mutuality

I note that the mutual fractal representation of the pairings may be employed to determine similar mutual representations of triplets, quadruplets, or larger groupings of images. As a notational convention, I construct these additional representations for triplets (M_{ijk}) and quadruplets (M_{ijkl}) in a like manner:

$$\mathbf{M_{ijk} = M_{ij} \cup M_{jk} \cup M_{ik}}$$

$$\mathbf{M_{ijkl} = M_{ijk} \cup M_{ikl} \cup M_{jkl} \cup M_{ijl}}$$

Thus, in a mutual fractal representation, there is the necessary apparatus for reasoning analogically about the relationships between images, in a manner which is dependent upon only features which describe the mutual visual similarity present in those images.

CHAPTER 3

FRACTALS AND KNOWLEDGE

In the preceding chapter, I illustrated the development and construction of the fractal representation. In this chapter, I specifically address the question of whether the fractal representation is a knowledge representation.

What is a Knowledge Representation

Acts of cognition involve the manipulation of knowledge, represented in some manner. While the term "representation" is quite commonplace and its use may be familiar, it is significant to note that very rarely is the notion tackled of what a representation actually may be. However, in the AI literature, a paper by Davis, Shrobe, and Szolovits, did address this issue (Davis et al., 1993), and Sowa later expanded on their criteria, albeit from a perspective of knowledge engineering (Sowa, 2000). Guarino (1995) also addressed the ontological aspects of representation.

The roles of representation

A representation can be said to have meaning when in service toward a particular task. Davis et al. (1993) note that representations play five distinct, critical roles. Those roles are as a surrogate, as a set of ontological commitments, as a fragmentary theory of reasoning, as a medium for pragmatically efficient computation, and as a medium of human expression. Let us consider each role in brief, and begin to bring aspects of visual search into the discussion.

Representation, as a surrogate

When a mind reasons about its world, this reasoning occurs internally, while the majority of what it reasons about exists externally. A representation then must act as a surrogate for things which exist outside the reasoning agency. Direct interaction with real world objects are paralleled by operations upon the internal representations of those objects.

Davis et al. (1993) raise two significant points concerning surrogates: what is a surrogate a surrogate for, and what is the fidelity of a surrogate? Some correspondence between the surrogate and its counterpart in the world must be specified. With respect to fidelity, what attributes of the original are preserved, omitted, or implied with the surrogate must be addressed, for perfect fidelity is impossible.

Representations, then, must be imperfect, and since reasoning operates upon representations, so to must reasoning itself arrive at imperfect conclusions, even if the reasoning process itself is sound. It is this correspondence aspect which must be adequately addressed in any system which seeks to concern itself with levels of abstraction.

Representation, as a set of ontological commitments

Selecting a representation involves a decision about how and what to represent from the arriving world. A set of commitments, then, is made that both define the extent of the representation's capture of the world and define the way that extent is expressed or embodied within the representation ontologically. Here, the task at hand acts as a guide toward the selection of an appropriate ontology. These commitments start at the moment a representation begins to form, and likely accumulate as the representation is used. As Davis et al. (1993) note, the representational power lies in the correspondence of the representation to something in the world and in the constraints that that correspondence impose.

Representation, as a fragmentary theory of reasoning

Representations are formed to allow cognition to occur within some agency. Even though the theory of reasoning arising from a representation may be implicit, it can be seen through three aspects: what the representation defines as inferencing, the set of inferences it allows, and the subset of those inferences which it recommends. I refer the reader to the Davis paper for a thorough discussion of what it is to make intelligent inferences.

Allowed inferences are those inferences which can be made from available information. As a representation might arise in any number of ways, so too might the allowed inferences vary. As Davis et al. (1993) point out, this flexibility is acknowledged so as to admit the legitimacy of the various approaches. Having this flexibility at its core provides a framework for re-representation.

Clearly, the set of allowable inferences may become untenably large. A smaller, constrained subset of these inferences is necessary. Whether by specifying the constraints with which to select recommended inferences, or by providing them somewhat explicitly, some process or reasoning or insight must be at work to frame them. In this way, Davis et al. (1993) citing Minsky by way of example, illustrates that representation and reasoning are intertwined in a deep, theoretical manner. They also observe that much of the reasoning which informs recommended inferences has been provided by observation of human behavior.

Representation, as a medium for efficient computation

The information processing stance of human cognition holds that cognition is a computational process. In the same sense that a representation recommends inferences, so to does it imply the manner in which it may be used in computation. This guidance speaks to the adequacy of the representation, as an organizational mechanism for information, for the task at hand.

Representation, as a medium of expression

Although the Davis paper addresses itself to the notion of representations as vehicles for human expression, I wish to stress that the internal dialogue of, about, and with representations is as important as the external one. In so complex a system as the human brain, information must pass from subsystem to subsystem, preferentially without substantial degradation and with increasing specificity. The expression of representations internally is a process of systematic reassembly of aspects of those representations into new ones, through which other systems may operate upon the newfound representations, with the core roles of representations implied by those systems' tasks. Herein, cognitive models are formed.

The representation definition and criteria of Markman

In his book "Knowledge Representation," Markman offers both a definition of representation as well as a set of criteria for assessing a representation (Markman, 1999). Let us first consider Markman's remarks.

Defining representation

Markman (1999) gives a definition of representation with four components. The four components are:

1. a represented world - the domain that the representations are about;
2. a representing world - the domain which contains the representations;
3. representing rules - a set of rules which map elements in the represented world to elements in the representing world; and
4. a process which uses the representation.

Markman notes that in all known representational systems, the representing world loses information about the represented world (Markman, 1999). Specifically, he assigns this loss of information to the decision made about what aspects of the represented world to be included in the representing world. That is, the agent constructing the representing world must decide what to include, and what to exclude, and that decision carries forward into the representation the consequences of it.

Markman notes that the representing rules determine the isomorphism (or homomorphism) of the representation: if each unique element in the represented world is mapped to a unique element in the representing world, the representation is isomorphic (Markman, 1999). The correspondence given by the representing rules also imply loss of information: if a representation is homomorphic, then more than one element in the represented world maps in an undifferentiable manner to the same element in the representing world, and therefore the ability to discriminate between those represented world elements is lost. This loss of information, through deliberate omission and through potential homomorphism, affords the capacity for reasoning about the missing information from that which is not missing.

Markman's requirement that the representation must be associated with some process which uses it implies that utility to an agent is the rationale for the construction of the representation. Markman additionally notes that Marr (1982) remarks that a given representation makes some information about the represented world easier to access than other information, via the representing rules and the loss of information.

Characterization

There are additional ways to characterize representations, and both Markman (1999) and Nersessian (2008) provide insights into how to achieve such characterizations.

Analog / Symbolic

Markman further distinguishes representations as either analog or symbol. A representation is an analog if the representing world has an inherent structure about how it operates and that the relationships between elements in the representing world are not arbitrary. A representation is symbolic if a convention exists which links all of the elements in the representing world, the convention being arbitrary in a sense that representing rules could be changed to determine a wholly new convention. In this way, the representing rules determine, Markman seems to suggest, the nature of a representation's analogism or symbolism.

Iconic / Propositional

Nersessian (2008), in a discourse on mental modeling, uses slightly different terminology to emphasize the same point. To Nersessian, a representation may be characterized as as iconic if it demonstrates a structural relationship to the thing it represents. Iconic representations therefore afford an ability to assess similarity or goodness of fit, and provide a notion of "accurate" or "inaccurate" (Nersessian 2008). A Nersessian iconic representation is thereby closely associated with Markman's analog representation.

In contrast, Nersessian holds that if the relationship between a representation and what it represents stands for a kind of "truth" and if the operations over the representation preserve this "truth" via the use of a consistent set of symbols which themselves stand for a stable collection of properties, then the representation is deemed propositional

(Nersessian, 2008). Therefore, Nersessian propositional representation is most closely aligned with Markman's symbolic representation.

Modal / Amodal

Nersessian (2008) further delineates representation along a dimension which pertains to the degree to which its symbols can be associated with perceptual states (Barsalou, 1999, 2008). Modal symbols are analog (in the Markman sense) representations of the perceptual states from which they are extracted. Amodal symbols, on the other hand, are arbitrarily (but consistently) assigned.

Therefore, in Nersessian's view, a propositional representation uses amodal symbols, but an iconic representation may use either modal or amodal symbols, or both.

Markman representational dimensions

Markman further suggests that proposed representations be assessed with respect to at least three dimensions: their endurance, the presence of symbols, and their abstractness. By endurance, Markman means not that some specific values within a representation be maintained (a state), but that the representation itself may be temporary or long-lasting. By the presence of symbols, this is a distinction between representations which are symbolic and which are not. Markman invokes the use of a space (as a structure upon which elements have some positional meaning) as an example of non-symbolic representation. Lastly, by abstractness, Markman suggests that this is the degree to which the process which uses the representation is distinct from the representation itself. Markman further develops the notion of the power of a representation as a convolution of another way in which to describe the suitability of the representation to the process which intends to use it with the expressivity of the representation (the degree to which it may be able to represent all represented worlds).

Fractal representations, in light of Markman

Let me now reflect on the fractal representation as a representation, working in somewhat the reverse order of the Markman definition and criteria.

Fractal representations and the represented / representing worlds

By beginning with the fractal encoding process, the fractal representation is a capture of that unordered set of transformations which transform one image into another image. The represented world is the set of the source and target images. The representing world is the set of transformations. There is a commitment, via the chosen partitioning scheme, as to what aspects of the represented world are selected as being contained in the representing world. Indeed, the partitioning scheme itself is the constructing agent's primary method by which the inclusion/omission of represented world information is made. The fractal representation satisfies this aspect of Markman's definition.

Fractal representations and the representing rules

Again, due to the use of the fractal encoding process itself in conjunction with the partitioning scheme, the representing rules clearly and distinctly map the represented world (source/target images) with the representing world (set of transformations). Moreover, this mapping is isomorphic, as the partitioning scheme must meet the connectivity and covering requirements described above. Therefore, the fractal representation satisfies this aspect of Markman's definition.

Fractal representations and symbolism

The fractal representation is non-symbolic in the Markman sense, in that it rests upon an inherent structure given by the representing rules which is non-arbitrary. However, as I point out above, the aspects of the fractal representation, the fractal features, may themselves by represented according to any suitable arbitrary convention, so long as they allow for discrimination between themselves. While the symbols chosen may indicate correspondence to certain non-arbitrary aspects of the representing world (a position in space, a color, etc.), the manner in which the features are denoted itself is independent and arbitrary with respect to inherent structure of the fractal representation and to the manner in which the comparison between features is made. Thus, fractal features are symbolic in the Markman sense, but the fractal representation from which they are derived are not. Even so, the fractal representation satisfies this aspect of Markman's definition.

The expressivity and power of fractal representations

The fractal representation is able to represent any two arbitrary real-world images. Furthermore, as developed in the section on mutual fractals, the fractal representation may be extended to represent any arbitrarily large set of images. Thus, the fractal representation affords a tremendous expressivity. But does this mean that it is a powerful representation? According to Markman, the power of a representation can only be determined via its suitability to some task. It is my belief that through the demonstration of the fractal representation's utility in addressing a wide variety of problems of visual similarity, visual oddity, and perception, I have provided reasonable evidence to suggest that the fractal representation is quite powerful with respect to those tasks.

But let me take it one step further. The expressivity and power of fractal representations is also rooted in the powerful association of the representation and the mathematical notions of iterated function systems. As seen in the prior chapter, iterated function systems have been used to characterize and to create models of a wide variety of physical and mathematical systems. The fractal representation provides a direct means to reconstruct the target image from its represented world if it is used in an iterated function system (IFS) manner; that is, a sufficient fidelity rendering of the target image may be obtained if the representation is used to calculate as an IFS, from any original image, for the target image is the encoded attractor. The power of the fractal representation stems not only from this aspect, but from the explicit use of the source image as the initial condition, forming a structural, spatial relationship between that initial condition and the attractor.

Lastly, the fractal representation is specifically and deliberately modal, for it expressly relates the received perceptual input of source and target images from the represented world and establishes an isomorphic mapping between that input and the representing world.

The knowledge representation roles of Davis et al.

Let me now address the roles of Davis et al. (1990) and fractal representations. The five roles are as a surrogate, as a set of ontological commitments, as a fragmentary theory of reasoning, as a medium for pragmatically efficient computation, and as a medium of human expression.

Fractal representation as a surrogate

Davis et al. (1990) argue that a knowledge representation is a surrogate for the world, over which reasoning is performed. A fractal representation is the representation of a pair (or more) of real-world images (source and target) as a finite set of similitude transformations. No reasoning about a fractal representation involves the original, represented world: reasoning is only performed on the set of transformations (or fractal features derived from them). The representation maintains a strong, direct correspondence between the represented and representing worlds, a consequence of the act of encoding and the choice of the partitioning scheme used by the encoding. Furthermore, the fidelity of the correspondence is determined precisely by the partitioning scheme. This commitment of the fractal representation to correspondence and fidelity, driven largely by the partitioning, allows the representation both to satisfy the first role as well as affords a powerful means by which the fidelity may be tuned, providing a different kind of abstraction (different from the Markman sense) which I develop in a subsequent chapter.

Fractal representation as a set of ontological commitments

The fractal representation, through the encoding process by which it is derived and the partitioning scheme which the encoding process uses to carve up the represented world, clearly makes a deliberate commitment and mapping between the represented and representing worlds. But is this ontologically sound?

My answer is yes. There is absolute grounding between each transformation in the representing world and the fragments derived from the represented world. Moreover, the mapping is wholly isomorphic. Thus, there is no other potential meaning for any of the transformations other than it precisely stands as the capture of the mapping between the fragments. Each transformation is complete, concise, and deliberately excludes implication or information from any other portion of the represented world. Thus, the fractal representation satisfies the second role.

Fractal representation as a fragmentary theory of reasoning

To consider the fractal representation as a fragmentary theory of reasoning, we must consider what the representation defines as inferencing, the set of inferences it allows, and the subset of those inferences which it recommends. As for inferencing, or in the broader sense of intelligent reasoning (ala Davis et al. 1990), I refer the reader to subsequent chapters of this dissertation, in which I explore in detail the fractal representation's suitability to addressing problems in visual similarity, visual oddity, and perception. But let us consider more closely the afforded and sanctioned "inferences" of the fractal representation.

The fractal representation clearly affords the ability to determine subsets of its core set of transformations. It also affords the determination of fractal features from each of these transformations, as well as the collection into strings or sets various collections of those features. It, at least in principle, admits the combination of aspects of its transformations into subsequent transformations, a topic which I explore in some detail at the end of this dissertation, under fractal composition.

The fractal representation specifically sanctions all of the above as well, but it does not, perforce, sanction the partial combination of aspects of one transformation with aspects of another. Why? Because to do so would negate the strong correspondence between the represented and representing worlds. Indeed, the only sanctioned operations are those which expressly maintains that correspondence.

Notice that a powerful operation that the fractal representation sanctions is the modification of the partitioning scheme used by the encoding process. This ability itself provides the mechanism by which a fractal representation can be rerepresented into a more coarse or more finer correspondence. This ability to shift levels of abstraction between the represented and representing worlds, afforded by modifying partitioning, is developed later in this dissertation.

All of the examples of reasoning using fractal representations contained in this dissertation specifically require this sanctioning. In this way, fractal representations satisfy the third role.

Fractal representation as a medium for pragmatically efficient computation

While process of encoding a pair of images fractally is computationally intensive, it is not to say that the resultant fractal representation itself carries that burden. In fact, one of the most common uses I make of the representation is to use fractal features as indices for storing or retrieving the representation in a memory. In this regard, the representation is phenomenally effective, as I demonstrate through the development and description of the Analogy by Recall algorithm subsequently. Furthermore, as I outline in the latter chapter of the dissertation, the ability to construct new fractal representations without resorting to recalculating the encoding is both computationally efficient and afforded and sanctioned. In this manner, the fractal representation satisfies the fourth role.

Fractal representation as a medium of human expression

This last role of Davis et al. proves the most vexing to argue for, for at its core, this would seem to require the communication of fractal representations between two agents in order to assess its expressivity. Let me tackle it in this way.

The fractal representation may need to be examined as a series of subsets of its original state, or rerepresented into a more coarse or more fine correspondence, in order for the agent to accomplish its task. The representation itself is far more compact than the original (and arguably infinite) data in the represented world. Thus, any subsystem which makes use of the representation or shares it with another (for example, from a memory system to a system which calculates featural similarity) benefits from the efficiency of this compaction.

So, is the fractal representation an effective means of human expression? Perhaps not, for I make no claim that what I have developed is a cognitive model (which is, after all, what Davis et al. (1990) address indirectly through the development of their knowledge representation roles. Instead, let me say that the fractal representation affords a profound computational model, clearly cognitively inspired.

The Fractal Representation is a Knowledge Representation

In light of the strength with which the fractal representation satisfies the criteria of Markman and meets the roles of Davis et al., I claim that, yes, the fractal representation is a knowledge representation. Furthermore, and to be quite specific ala Markman and Nersessian, the fractal representation is an analog/iconic modal knowledge representation.

CHAPTER 4

FRACTALS AND VISUAL SIMILARITY

This chapter will discuss visual similarity, and a class of problems from visual analogy in which similarity calculations are used to derive an answer. This chapter serves to introduce the Analogy By Recall (ABR) algorithm.

Visual Analogy and Similarity

Suppose there is a visual analogy, expressed symbolically as A : B :: C : D, with the symbols representing images, as shown in Figure 4.1. This can be interpreted as suggesting that some operation T exists which captures the relationship between image A and image B ("A is to B"). Likewise, some other operation T' is proposed which captures the relationship between image C and image D ("C is to D").

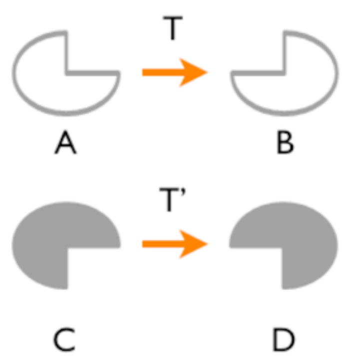


Figure 4.1. An Example of Visual Analogy

In this manner, it is seen that the central analogy in such a problem rests not with the images themselves, but in the degree to which the two operations T and T' are analogous or similar. I can express the problem to make plain this distinction thus:

$$\mathbf{A : B :: C : D \;\; \longrightarrow \;\; T(A,B) :: T'(C,D)}$$

Similarity between operations

The nature of this similarity may be determined by a number of methods, many of which might associate visual or geometric features to points in a coordinate space, and compute similarity as a distance metric. Tversky developed an alternate approach by considering objects as collections of features, and similarity as a feature-matching process (Tversky, 1977).

I adopt Tversky's interpretation of similarity, and thus seek to express these operations T and T' in some representation which both is robust and affords sufficient feature production to permit feature-matching (Ashby & Ennis, 2007). A particular nuance of Tversky's approach, however, is that either the representation or the features derived from the representation must be formable into sets, as the calculation for similarity employed requires the counting of elements within sets (and their union and intersection).

Thus, I can revisit the typical visual analogy A : B :: C : D, where T and T' are now representations which meet Tversky's featural requirement. To make a comparison between the two representations, I first derive features from each, and then calculate a measure of similarity based upon those features.

Similarity metric

I desire a metric of similarity which is normalized, one where the value 0.0 means entirely dissimilar and the value 1.0 means entirely similar. Accordingly, I use the ratio model of similarity as described by Tversky (1977), wherein the measure of similarity between the two representations T and T' is calculated thus:

$$\mathbf{S(T,T') = Ƒ(T \cap T') / [Ƒ(T \cap T') + \alpha\, Ƒ(T\text{-}T') + \beta\, Ƒ(T'\text{-}T)]}$$

where the operator Ƒ(Y) derives the number of features in some set Y. The particular sets involved may be considered as indicating, respectively, those features the two representations share ($T \cap T'$), those features in T but not in T' (T-T'), and those features in T' but not in T (T'-T).

Tversky (1977) notes that the ratio model for matching features generalizes several set-theoretical models of similarity proposed in the psychology literature (e.g. (Bush & Mosteller, 1953) and (Gregson, 1976)), depending upon which values one chooses for the weights α and β. Later in this discussion, I shall revisit these weights, and illustrate the significance of their choice.

A Strategy for Visual Analogies

One can interpret visual analogies as suggesting that some operation T exists which captures the relationship between image A and image B ("A is to B"). Likewise, some other operation T' is proposed which captures the relationship between image C and image D ("C is to D"). Let us now consider a class of visual analogy puzzles, an example of which is shown in Figure 4.2.

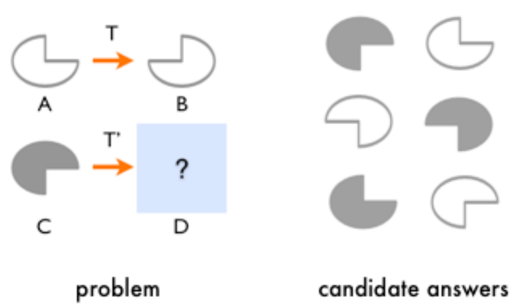


Figure 4.2. A visual analogy puzzle.

In this problem, the image D is missing, and the challenge is to determine which of the offered candidate images would best fit into the matrix. That is, it must be determined which of these candidate images, if selected as image D, would establish transformation T' as most analogous to transformation T.

Analogies in a general sense are based on similarity and repetition (Hofstadter, 2008). I would seek to employ a suitable representation, one which affords the capture of these qualities as well as sanctions reasoning over them. As I showed previously, fractals capture self-similarity and repetition at multiple scales (Mandelbrot, 1982), and I therefore propose that fractal representations are an appropriate choice for addressing certain classes of analogy problems.

One method for solving this puzzle is this: from this set of candidates, form the fractal representations from the fractal encoding of the transformation of each candidate image X in terms of image C.

$$\forall\, \mathbf{X} \in \{\ \textbf{candidate answers}\ \},\ \mathbf{T_x} \coloneqq \textbf{FractalEncode(C,X)}$$

$$\mathbf{\Omega = \{\ T_1, T_2, T_3, T_4, \ldots\ T_n\ \}\ \textbf{and}\ T' \in \Omega}$$

This provides a set of possible transformations, which I shall label Ω, from which to seek the most analogous transformation T' and thereby find which candidate image was responsible for it.

The Analogy By Recall (ABR) Algorithm

I claim that analogy initiates with an act of being reminded, and that fractally representing both that triggering percept as well as all prior percepts affords unprecedented similarity discovery, and thereby analogy-making. I have developed and implemented an algorithm, called Analogy By Recall (ABR), to assist in illustrating and refining these claims.

The Generality of Representations

While in this dissertation I exclusively shall use fractal representations in the examples and subsequent discussion, the overall approach is agnostic with respect to representations, and may be used with any representation which affords the ability for objects thus represented to be decomposed into a set of features. The approach is distinguished from other analogical algorithms in that it presumes no explicit relationship between objects or between features of objects.

Introducing the Analogy By Recall (ABR) Algorithm

My approach compares each transform in the set Ω to the original transform T by means of recalling common features and calculating similarity metrics. This method is divided into several stages. I now present the algorithm in pseudo-code form, and then describe each stage of the algorithm in detail.

To determine the transform T' which is most analogous to transform T from a set of transformations $\Omega \coloneqq \{ T_1, T_2, T_3, T_4, \dots T_n \}$:

PREPARATORY

Let $\Omega^* \coloneqq \{ T \} \cup \Omega$
Construct a memory M as an empty hash table.
Let F() be a function which generates a set of features.
Let K() be an injective hash function for M.

INDEXING

For each transform $\tau \in \Omega^*$, hash τ in M by:
· Generate a set of features $F(\tau) = \{ f_1, f_2, f_3, \dots \}$.
· For each feature $f_j \in F(\tau)$, store τ into M, using $K(f_j)$ as a key.

RETRIEVAL

For each transform $T_i \in \Omega$, calculate S_i as the similarity of T to T_i by:
· Set $a \leftarrow b \leftarrow c \leftarrow 0$.
· Generate a set of features $F(T_i) \coloneqq \{ f_1, f_2, f_3, \dots \}$.
· For each feature $f_j \in F(T_i)$:
  - Use $K(f_j)$ as a key to retrieve a set of entries μ from M.
  - If $T \in \mu$, then $a \leftarrow a + 1 \because f_i \in F(T_i) \cap F(T)$.
  - If $T \notin \mu$, then $c \leftarrow c + 1 \because f_i \in F(T_i) - F(T)$.
· Generate a set of features $F(T) \coloneqq \{ f_1, f_2, f_3, \dots \}$.
· For each feature $f_j \in F(T)$:
  - Use $K(f_j)$ as a key to retrieve a set of entries μ from M.
  - If $T_i \notin \mu$, then $b \leftarrow b + 1 \because f_i \in F(T) - F(T_i)$.
· Calculate S_i from the values a, b, and c:

$$S_i \leftarrow a / (a + \alpha{*}b + \beta{*}c)$$

Determine $\zeta \leftarrow \max \{ S_1, S_2, S_3, S_4, \dots S_n \}$

T' is therefore that transform $T_i \in \Omega$ which corresponds to the maximal similarity ζ, and is deemed the most analogous to transform T.

Algorithm 4.1. The Analogy by Recall (ABR) Algorithm.

Analogy by Recall: Preparatory Stage

My system uses a feature-based similarity approach to analogy. Consequently, I chose data structures which facilitate the storage and retrieval of information based upon aspects of the data, specifically by using a hash table as a data structure surrogate for memory. As transformations will be hashed into memory, I define two additional operators: F(), a method to generate a set of features from a given transformation; and K(), an injective hash function which operates solely over the domain of the features.

I made the commitment to a hash table for two reasons beyond that of wishing to use features. First, it is desirous to find some overlap in the features which occur between two transformations, such that a perfect overlap would deem the transformations perfectly analogous. The hash function K() may result in hashing multiple transformations to the same feature, and therefore K() must operate only upon a given feature, and not take into consideration the transformation which gave rise to that feature. Second, F(), the method which generates features from a transformation, must do so in a manner such that each generated feature affords salience, or information content (Tversky 1977).

Analogy by Recall: Indexing Stage

I wish to store each transformation in the hash table memory M. The set of possible analogous transformations Ω is combined with the original transformation T to form a new set Ω^*. The algorithm iterates over each member $\tau \in \Omega^*$, and from each member calculates a set of features using $F(\tau)$. For each feature $fi \in F(\tau)$, the transformation is indexed as an ordered pair $(K(f_i), \tau)$. That there likely will be hash collisions at key value $K(f_i)$ is expected and desired.

Analogy By Recall: Retrieval Stage

A measure of similarity between the original transformation T and each possible analogous transformation $Ti \in \Omega$ must be determined. The choice of metric reflects similarity as a comparison of the number of features shared between candidate pairs taken in contrast to the joint number of features found in each pair member (Tversky, 1977). I desire a metric which is normalized with respect to the number of features under consideration. In my implementation, the measure of similarity between the target transform T and a candidate transform T_i is calculated using the ratio model (Tversky, 1977):

$$\mathbf{S(T,T_i) = Ƒ(T \cap T_i) / (Ƒ(T \cap T_i) + \alpha\, Ƒ(T\text{-}T_i) + \beta\, Ƒ(T_i\text{-}T))}$$

and Ƒ(Y) is a function which determines the number of features which may be extracted from the set Y. These values may be calculated effectively, using hash table retrieval as a surrogate for distinguishing and counting common and distinct features within the sets $T \cap T_i$, $T\text{-}T_i$, and $T_i\text{-}T$ respectively.

Tversky notes that the ratio model for matching features generalizes several set-theoretical models of similarity proposed in the psychology literature, depending upon which values one chooses for the weights α and β (Tversky, 1977). I have found that significant discrimination between candidate answers may be found by using the Jaccard similarity; that is, by setting $\alpha \leftarrow \beta \leftarrow 1.0$, and thus favoring features from either transformation equally. As Tversky (1977) notes, by equating α and β, I ensure that the calculation of similarity is symmetric with respect to the transformations under comparison.

Once the algorithm has calculated the similarity function over all of the candidate transforms, it is a straightforward matter to determine which transformation has generated the maximal similarity. This transformation, T’, is deemed to be the most analogous to the original transformation T.

An Example

I now present an example of using fractal representations and my strategy to solve the visual analogy puzzle shown in Figure 4.2 above.

The primary and candidate transformations

In this example, the problem is to determine for which of the candidate images the transformation T' is made most analogous to transformation T. I first will represent T as a fractal representation, and then generate a set of candidate transformations Ω as shown in Figure 4.3.

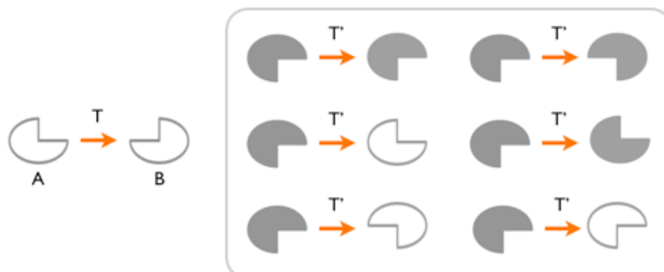


Figure 4.3. The primary and candidate transformations

I arbitrarily may select any partitioning scheme, so long as that partitioning meets the criteria of coverage outlined above. For the purpose of this example, let us choose to partition each image into a series of 16x16 pixels, forming a regular grid. Each of these images is 134 pixels wide and 84 pixels high. Thus, each image is to be partitioned into 54 blocks.

Each of these blocks will be represented by a single fractal code. In my present implementation, each fractal code generates 63 features. Therefore, at this partitioning, each primary transformation will be indexed into memory using the 54 x 63 x 2 = 6804 features generated from mutual fractal representation of that transformation.

Calculating similarities and selecting the most analogous.

After the primary transformation T is indexed into memory using features derived from the fractal codes, a similarity value for each of the candidate transformations in the set Ω may be calculated using the Tversky formula as noted. Table 4.1 illustrates the values calculated for each of the candidate transformations.

Table 4.1. *Candidate transformation similarities*

transformation	similarity
	0.623750735
	0.517048795
	0.552910053
	0.84244562
	0.573780129
	0.535273369

It may be seen that the fourth transformation is the most similar to the primary transformation T, with a value of 0.842. Therefore, for this puzzle, the answer is candidate answer 4.

Confidence

In the Analogy by Recall algorithm, and as illustrated by the example above, a candidate representation can be found to be the most analogous by a straightforward calculation of its featural similarity, using the Tversky formula. But a question quickly arises: how confident is the answer? That is, given the variety of answer choices, even though an answer may be selected based on maximal similarity, how may that choice be contrasted with its peers as the designated answer?

A potential path forward would be to assess the given similarity calculation as a member of the set of all such similarity calculations for the collection of potential answers. Let us suppose that for a given collection of answers, a strategy such as ours is used to calculate a corresponding set of similarity values:

$$\mathbf{ABR(\ \{A_1, A_2, A_3, A_4, \ldots A_n\ \},\ problem\ ) \rightarrow \{\ S_1, S_2, S_3, S_4, \ldots\ S_n\ \}}$$

$$\mathbf{\forall S_i,\ 0.0 \leq S_i \leq 1.0}$$

The ABR algorithm would offer

$$\boldsymbol{\zeta} \leftarrow \mathbf{max(\ \{\ S_1, S_2, S_3, S_4, \ldots\ S_n\ \}\ )}$$

as the maximal similarity value, and thereby deem the answer which generated that value as the most analogous. It may be determined, additionally, how statistically distinct the value ζ is from its peers, by first calculating the mean, standard deviation, and standard error of the mean for the set of similarity values, and then, assuming a normal distribution, calculating the deviation of each of the values from the mean of the set:

$$\boldsymbol{\mu = n^{-1} \Sigma\ S_i \quad \text{and} \quad \sigma = \sqrt{\ [n\text{-}1\ \Sigma\ (S_i\text{-}\mu)^2]} \quad \forall i,\ 0 < i\ \leq\ n}$$

$$\boldsymbol{\sigma_\mu = \sigma/\sqrt{n}}$$

then

$$\mathbf{Deviations} \leftarrow \mathbf{\{\ D_1, D_2, D_3, D_4, \ldots\ D_n\ \}}$$

$$\boldsymbol{\forall i,\ 0 < i\ \leq\ n,\ D_i = (S_i\text{-}\mu)/\sigma_\mu}$$

where the set Deviations is a t-distribution of the similarity values. The most analogical answer, the one corresponding to the maximal similarity value ζ, would have the largest positive deviation value under this reformulation:

$$S_x = \zeta \leftarrow \max(\{ S_1, S_2, S_3, S_4, \ldots S_n \}) \text{ iff}$$

$$\exists\, x = y, D_y = \max(\{ D_1, D_2, D_3, D_4, \ldots D_n \})$$

This, then, suggests that the most analogical answer would in a sense "stand apart" from the rest of the answers. The degree to which it "stands apart" may be interpreted as a metric of confidence in selecting the answer. Indeed, assuming a normal distribution, a confidence interval based upon the standard deviation may be calculated, and score each of these values along such a confidence scale, where 0.0 would indicate no variation at all from the answer, and 1.0 would indicate an utterly apparent and distinct value. Thus, the problem of selecting the most analogous answer is transformed into a problem of distinguishing which of the possible answers is a statistical outlier.

Ambiguity

The similarity scores generated by the ABR algorithm may vary widely. As shown above, the problem of selecting the most analogous answer may be considered as the companion problem of determining the statistical outlier. The challenge is that there may be more than one such outlier, or none at all. I deem these situations ambiguous.

To resolve such ambiguity, one must first examine why such a situation might arise. I argue that the ambiguity arises due to a data problem, but it is more: it is a problem with the representation itself, from whence the data arise. The similarity value calculated by the ABR algorithm is determined by the Tversky formula for similarity:

$$Si \leftarrow S(T,Ti) = Ƒ(T \cap Ti) / (Ƒ(T \cap Ti) + \alpha\, Ƒ(T-Ti) + \beta\, Ƒ(Ti-T))$$

which is itself wholly dependent upon Ƒ(), the number and nature of the features being considered, and the intersection and difference of the sets of those features.

Homogeneity and Sparsity

Therefore, it is the homogeneity of the features (those which occur in both sets, Ƒ(T∩Ti)), and the sparsity of the number of features, which directly affect the similarity calculation. These two factors in turn affect the ability of the ABR algorithm to offer an unambiguous selection of the most analogous answer.

To address the sparsity of data, there must be determined a way to create more of it. To address homogeneity of data, the manner in which the data is created must be modified, so as to afford potential variance. In either case, what is sanctioned by the representation over which the analogies are being formed must be examined.

Resolving Ambiguity

As noted in Chapter 3, Davis et al. (1993) describe the five distinct roles that representations play: as a surrogate, as a set of ontological commitments, as a fragmentary theory of reasoning, as a medium for pragmatically efficient computation, and as a medium of human expression. Even though the theory of reasoning arising from a representation may be implicit, it can be seen through three aspects: what the representation defines as inferencing, the set of inferences it allows, and the subset of those inferences which it recommends.

Allowed inferences are those inferences which can be made from available information. As a representation might arise in any number of ways, so too might the allowed inferences vary. As Davis, et al. (1993), point out, this flexibility is acknowledged so as to admit the legitimacy of the various approaches. Having this flexibility at its core provides a framework for re-representation.

However, the set of allowable inferences may become untenably large. A smaller, constrained subset of these inferences is necessary. Whether by specifying the constraints with which to select recommended inferences, or by providing them somewhat explicitly, some process or reasoning or insight must be at work to frame them.

In the same sense that a representation recommends as well as sanctions inferences, so to does it imply the manner in which it may be used in computation. According to Davis et al. (1993), it is this guidance which speaks to the adequacy of the representation, as an organizational mechanism for information, for the task at hand.

Sanctioned operations on fractal representations

Performing reasoning afforded by the fractal representation of the relationship between images limits the mechanisms to those which the representation sanctions. There are two primary sanctions of the representation: the number of fractal codes which constitute the representation, and the creation of features from those fractal codes. These two sanctions offer methods by which the data problem of ambiguity that arises in analogical reasoning by recall may be addressed.

Fractal Abstraction

The features available from a fractal representation are derived from that representation's constituent fractal codes. The number of fractal codes in a particular fractal representation is determined solely by the partitioning scheme chosen when constructing the representation. The twin key observations of images which entailed fractal encoding (repetition and similarity at different scales) may be exploited here. In essence, partitioning is a modeling of how coarsely or finely an image is received or regarded, and that granularity determines the algorithm's ability to capture within the representation any present repetition or inherent similarity at that limit. Thus, the partitioning affords a level of visual abstraction.

Increasing the degree of partitioning accomplishes two acts: more fractal codes are created, and the possible variety of features arising from those codes increases. Both of these may address ambiguity in the data.

However, a closer consideration reveals further nuances in abstraction. As the partitioning becomes finer, there likely is a level at which the ambiguity is resolved. However, as the partitioning surpasses that point, and becomes finer, the answer may well become ambiguous once again. What does this suggest with respect to a balancing between the sparsity of data (the number of available features) and the homogeneity of data?

Emergent Sufficient Abstraction

As the level of abstraction becomes finer (resolution increases due to increased partitioning), the number of fractal codes, and thereby the number of features, rises. As resolution increases, the fractal codes represent partitioned areas in the image that are covering ever smaller areas. These areas become increasingly more homogenous, and therefore the fractal codes become more similar to one another (that is, their features become more consistent). As the abstraction grows finer, more codes are devoted to representing areas of consistent color and texture. Even though the number of codes and features is increasing, the homogeneity, and thereby the ability to discriminate based on those features, is decreasing. I believe this equates to a frequency apprehension of the image, with coarse resolution corresponding to low frequencies (fundamentals), and fine resolution corresponding to high frequencies (overtones, and then noise).

Thus, the disappearance and reemergence of ambiguity is an emergent characteristic. This emergent sufficient abstraction suggests that the Analogy by Recall algorithm may be modified to notice this characteristic automatically, adopted as a meta-reasoning strategy, sanctioned by the representation. In doing so, this strategy expresses the first aspect of visual perception: the relationship between the observer and the observed.

Ambiguity and Mutuality

Above, I illustrated how ambiguity may be resolved through re-representation via adjusting the level of abstraction (or degree of partitioning), and that a strategy may be derived which notices the need for such repartitioning in an automatic fashion. There exists a case which bears brief further discussion: what if every level of detail or repartitioning results in continued ambiguity?

In the case of resolving ambiguity through changing abstraction, the representation from which the features are derived remains one in which the relationship between objects remains fixed. Let us examine the situation where the relationship itself may vary.

I described earlier the notion that a fractal representation may capture not just the relationship between two images, but that it may be extended to describe the relationship between an arbitrary number of images. This is attributable specifically to the nature of the fractal representation itself: it is an unordered set of fractal codes, from which features are derived.

Depending upon the analogy problem, it may be feasible to consider first analogies between pairs of images. Then, should ambiguity be manifested at all practical levels of abstraction for those pairs, the algorithm could shift to consideration of triplets, quadruplets or other groupings of images, until such a time as ambiguity may be resolved. This shifting to higher order relationships, expressed by complexity in groupings, as a strategy for meta-reasoning, is expressly sanctioned by the use of fractal representations.

The ABR* Algorithm

Given these two sanctioned meta-reasoning strategies, I now revisit the ABR algorithm and extend it to incorporate them in an autonomous fashion. This extended algorithm I call the ABR* algorithm.

To determine the transform T' which is most analogous to transform T from a set of transformations $\Omega \coloneqq \{ T_1, T_2, T_3, T_4, \dots T_n \}$:

PREPARATORY

Let $A := \{ 1, 2, 3, \dots n \}$ represent an ordered range of abstraction
Let $G := \{ 1, 2, 3, \dots m \}$ represent an ordered range of complexity groupings
Let E be a real number which represents the number of standard deviations beyond which a value's answer may be judged as "confident"

EXECUTION

For each complexity $g \in G$:
  For each abstraction $a \in A$:
- Re-represent T' and Ω according to g and a
- Derive the set of similarity values $S := \{ S_1, S_2, S_3, S_4, \dots S_n \}$ by way of the ABR algorithm
- Set $\mu \leftarrow \text{mean}(S)$
- Set $\sigma_\mu \leftarrow \text{stdev}(S)/\sqrt{n}$
- Set $D \leftarrow \{ D_1, D_2, D_3, D_4, \dots D_n \}$ where $D_i = (S_i-\mu)/\sigma_\mu$
- Generate the set $C := \{ C_i \dots \}$ such that $C_i \in D$ and $C_i > E$
- If $|C| = 1$, return T' as the transform $T_i \in \Omega$ which corresponds to C_i
- otherwise there exists ambiguity, and further refinement must occur.

If no answer has been returned, then no answer may be given unambiguously.

Algorithm 4.2. The Extended Analogy by Recall (ABR) Algorithm.*

Note that as presented above the ABR* algorithm suggests that abstraction be increased before complexity grouping. This is not a strict guideline, and the inner and outer loops may be interchanged without loss of generality. Note also that it is possible that the algorithm will be unable to return an answer (if ambiguity fails to be resolved). In that case, the normal ABR algorithm may be used to choose the answer with the maximal similarity metric at a designated abstraction and complexity. Finally, the value E, which is used to designate a confidence level, may itself be varied as a meta-reasoning strategy.

The Example, Revisited

I return to my earlier example, now with the notion that using the ABR* algorithm may vary the level of abstraction to hone into a particular answer with confidence.

Determining a range of abstraction

In the ABR* algorithm, the partitioning scheme will vary, with the intention of automatically arriving at the appropriate level of abstraction at which the most suitable answer image may be selected with some confidence. To do so, a range of values must be established for the partitioning and a manner by which the partitioning will be refined at each subsequent step as necessary.

Let us establish a finest level of possible detail as a grid size of 2 x 2, as I noted earlier that such a size is the smallest which affords the set of admissible similitude transformations. The intent is to systematically reduce from a coarsest level of detail to this finest level. In my present implementation, as it resolves in to ever finer levels, the grid size is halved. Therefore, using this strategy of resolutions, for an image with a maximal pixel dimension of N, this formula determines the coarsest level of detail:

$$\mathbf{coarsestLevel} \leftarrow \mathbf{2}^{(1 + \lfloor \log 2\ N \rfloor)}$$

The images in the example are 134 pixels width by 84 pixels high, stored in the .PNG format. Thus, images with a maximal pixel dimension of 134, the coarsestLevel will be equal to 256. Using a strategy of halving the grid size, and progressing through each level of detail, this will provide for examination of the problem at 8 levels of detail. Each of the 134 x 84 pixel images will be placed into the center of a new 256 x 256 empty image, and it is these new images from which the ABR algorithm will commence.

Choosing a level of confidence

Since the set of deviations determined is a t-distribution of the similarity values, the choice of a level of confidence in selecting the answer can be made using conventional statistics.

If a normal distribution of the deviations is assumed, then an interval may be devised such that with an expected confidence of C for any deviation d, the probability of d in that interval is C:

$$\mathbf{P(-z \leq d \leq z) = C}$$

Note that this also lets one say that that the probability of d outside of the interval [-z,z] is:

$$\mathbf{P(d < -z) = P(d > z) = \tfrac{1}{2}(1.0 - P(-z \leq d \leq z)) = \tfrac{1}{2} (1.0 - C)}$$

The set of deviations is a t-distribution of the similarity values. If this is a normal distribution:

$$\mathbf{P(-z \leq d \leq z) = \Phi(z) - \Phi(-z) = erf(z / \sqrt{2})}$$

$$\mathbf{z = \sqrt{2}\, erf^{-1}(C)}$$

where Φ() is the cumulative normal distribution function and erf() is the error function. Therefore, given some value C, the equation above will determine the boundary.

Let us suppose, as an example, that for C = 90%, some answer X_i is the most analogous one. This would imply that the probability of that answer's corresponding deviation D_i obeys this:

$$\mathbf{z = \sqrt{2}\, erf^{-1}(0.90) = 1.644853}$$

Thus, if the deviation D_i is larger than 1.644853, with 90% confidence, the answer X_i is the most analogous.

For this example, let us chose a 95% confidence value, therefore seek a deviation which is larger than 1.959964, approximately a 2-sigma signal.

Calculating deviations and selecting an answer with confidence

In Table 4.2, I present the deviation values found for all eight levels of abstraction. Note that even at the coarsest levels, the correct answer to the puzzle (transformation #4) is significantly more deviant from the mean. But the information in the table warrants further consideration.

Table 4.2. Candidate transformation similarities

T'	deviations							
	0	-1.46	0.60	0.55	0.33	3.23	3.71	4.39
	-2.24	-0.97	-1.81	-1.73	-1.83	-2.19	-2.26	-1.89
	0	0.03	-1.03	-1.66	-1.11	-1.81	-1.77	-2.04
	4.47	4.87	4.68	4.71	4.76	2.93	2.09	0.712
	0	-1.48	-0.79	-0.84	-0.68	-0.56	0.12	0.36
	-2.24	-0.97	-1.65	-1.03	-1.47	-1.60	-1.90	-1.54
grid size	256	128	64	32	16	8	4	2
μ	0.492	0.283	0.307	0.298	0.606	0.684	0.758	0.866
σ_μ	0.116	0.095	0.098	0.085	0.049	0.038	0.027	0.013
codes	2	4	12	30	108	374	1428	5628
features	126	252	756	1890	6804	23562	89964	354564

Abstraction and Ambiguity

My experimental data suggests that there exists states of abstraction of a representation at which ambiguity vanishes, and others for which ambiguity is present. A strategy for determining an appropriate level of abstraction might entail first discovering those conditions at which there are marked changes in ambiguity. In earlier sections, I established the connection between ambiguity and abstraction: I now formalize the relationship, and explore how to conduct this discovery of ambiguity changes.

Ambiguity as a function

Ambiguity might be modeled expressly as a function of abstraction. In so far as the level of abstraction could be considered as a continuum from most coarse to most fine given a particular representation, this function also may be viewed as continuous.

Then would ambiguity as a function be thus:

$$\boldsymbol{\xi = f(\ a\ ),\ \forall\ a \in A,}$$

where A is the set of abstractions derivable from some representation R.

Note that here I refer to some representation R. Even those I have previously considered the problems and examples in this dissertation as separate representations, I now wish to regard R as the summation of those individual representations. For example, one may say that R is the set of representations that encompass the candidate solution images for a matrix problem. I presume here to seek those levels of abstraction which cause one aspect of R to be distinguishable from all other aspects of R.

With ambiguity expressed as a function, the boundary conditions of abstraction may be regarded mathematically as those places at which the function achieves a local minima or maxima. For convention, let us denote an ambiguity value of 0 as a best possible local minima; that is, for an ambiguity value of 0, the level of abstraction is such that an aspect of R is completely distinct from all other aspects of R.

The emergence of abstraction boundaries

The deviations presented in table 4.2 appear to suggest that if one starts at the very coarsest level of abstraction, the answer is apparent. Additionally, it seems to suggest that if one starts at the finest level of abstraction, another quite different answer is apparent. Both of the deviations for these levels, 4.47 for the coarsest and 4.39 for the finest, are unique for the set of answers at those levels, and deviations of that magnitude would suggest confidence levels of >99.99% if a normal distribution of error is presumed.

Extrema in data

I propose that in both cases, at the extrema of abstraction, the ABR* algorithm is operating with either too sparse a data set (at the coarsest) or with too homogeneous a data set (at the finest). Indeed, one can see that at the coarsest abstraction, there are 126 features upon which to calculate similarity, and at the finest abstraction, there are more than 350k features.

The data in the table offers the possibility of automatically detecting these situations. I suggest that the average similarity measurement should increase as the number of features against which it is calculated increases. Yet, one can see that the average similarity measurement at the coarsest abstraction is 0.492, but then falls, at the next level of abstraction, to 0.283, only to thereafter generally increase. I claim this constitutes an emergent boundary for coarse abstraction.

Other shifts

There exist other changes of note within the data. The average similarity measurement abruptly shifts value between grid sizes 32 and 16. In addition I specifically observe the arrival of ambiguity for grid sizes 8 and 4, in the sense that no value achieves the sufficient level of confidence required to make an answer.

Sufficient Abstraction

My interpretation is that the emergent sufficient level of abstraction, then, is at a grid size of 16, at that place where there appear to be a substantial number of features available for reasoning and yet those features retain discriminatory power that finer abstraction levels lack.

The Interplay of Observer, Observed, and Context

To deem some apprehended object as similar or novel involves the complex interplay of at least two relationships (Wagemans et al, 2012a and 2012b): the relationship between the observer and the observed, and the relationship between the observed and its context. The relationship between the observing agent and the observed object may vary depending upon some act taken by the observer. For example, if one wishes to appreciate an object at a higher level of detail, one might move closer to the object, or bring the object closer, resulting in the object occupying a larger expanse of the observer's field of view. This action modifies the resolution of the object: at differing levels of resolution, fine or coarse details may appear, which may then be taken into the consideration of the novelty of the object. The observed object also is appreciated with regard to other objects in its environment. Comparing an object with others around it may engage making inferences about different orders of relationships. The comparison may begin at a lower order but then proceed to higher orders if needed. The context also sanctions which aspects, qualities, or attitudes of the objects are suitable for comparison.

Analogies in a general sense are based on similarity and repetition (Hofstadter, 2008), and so I hold that fractal representations are a suitable representation, one which affords the capture of these qualities as well as sanctions reasoning over them.

The strategies employed in the Extended Analogy by Recall (ABR*) algorithm address both aspects of similarity and novelty detection I described above. It models the relationship between the observer and the observed by starting with fractal representations encoded at a coarse level of resolution, and then adjusts to the right level of resolution for addressing the given problem. It models the relationship between the observed and its context by searching for similarity between simpler relationships, and then shifts its searches for similarity between higher-order relationships. In each aspect, these adjustments are made automatically by the strategy of ABR*, by characterizing the ambiguity of a potential solution.

CHAPTER 5

FRACTALS AND RAVENS

Raven's Progressive Matrices Test suite is a set of standard and common tests of intelligence (Raven et al. 2003). The standard version of the test consists of 60 geometric analogy problems. Figure 5.1 illustrates a problem typical to those that appear on the test. Throughout this dissertation, when I refer to Raven's problems, I use example problems which are similar to those found on Raven's tests, due to copyright concerns and to ensure the integrity of the tests themselves. The results I report below, however, are from the actual test problems.

The task in the problem is to pick one of the eight choices in the bottom of the figure for insertion in that bottom-right element of the 3x3 matrix in the top of the figure. The chosen element should best match the patterns in the rows and columns of the matrix.

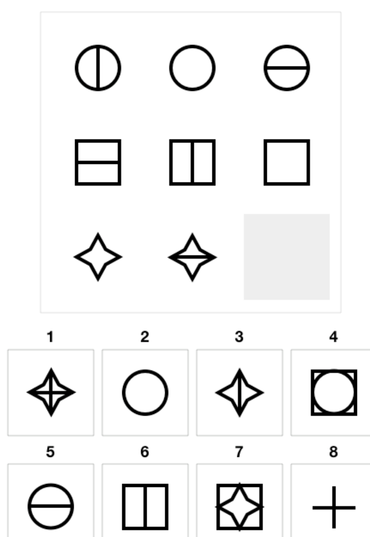


Figure 5.1. Problem similar to those of the Raven's Standard Progressive Matrices test.

The Raven's Progressive Matrices (RPM) test paradigm is intended to measure eductive ability, the ability to extract and process information from a novel situation (Raven et al. 2003). Eductive ability stands in contrast to reproductive ability, which is the ability to recall and use previously learned information.

The problems from Raven's various tests are organized into sets. Each successive set is generally interpreted to be more difficult than the prior set. Some of the problem sets are 2x2 matrices of images with six possible answers; the remaining sets are 3x3 matrices of images with eight possible answers.

The tests are purely visual: no verbal information accompanies the tests. The test-taker is asked to select from the available possible answers the single answer that best completes the matrix (Raven et al. 2003).

A Raven's Example

Let us illustrate the use of fractal representations and the ABR* algorithm for solving Raven's matrices problems. I shall use as an example the 3x3 matrix problem shown above.

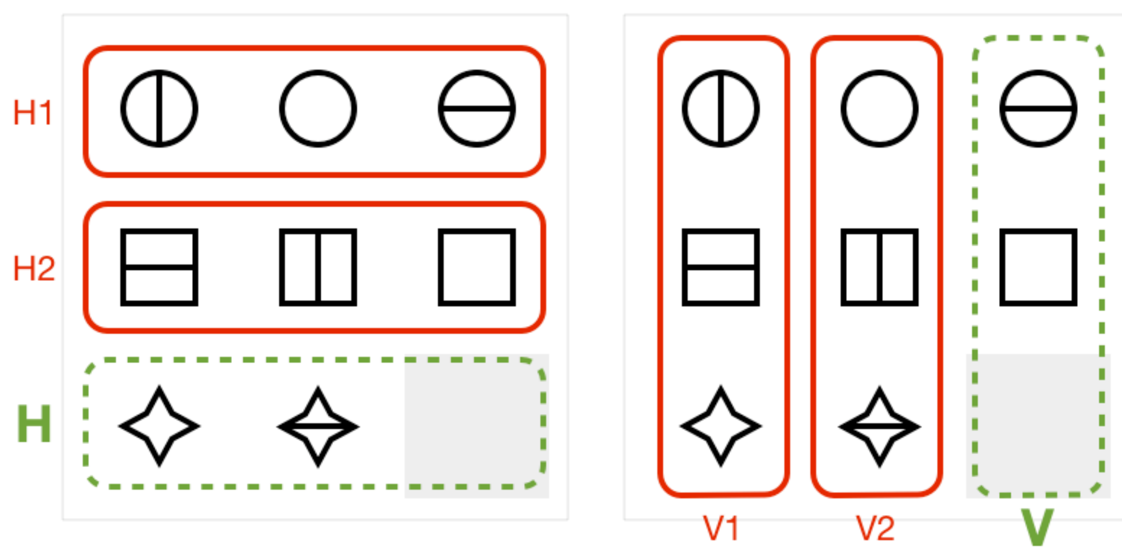


Figure 5.2. The simultaneous relationships

Simultaneous Relationships, Multiple Constraints.

An aspect of any Raven's problem, whether 2x2 or 3x3, is that there exist simultaneous horizontal and vertical relationships which must be maintained by the selection of the most analogous answer. In a 2x2 problem, there is one horizontal and one vertical relationship which constrain the selection. In a 3x3 problem, there are two horizontal and two vertical relationships. In my implementation, I represent these relationships as mutual fractal representations.

In Figure 5.2, I illustrate these relationships using the example problem. As shown, relationships H1 and H2 constrain relationship H, while relationships V1 and V2 constrain relationship V. There are other possible relationships which can be suggested by this problem: I have chosen to focus on these particular four relationships for clarity.

To solve a Raven's problem, one must select the image from the set of possible answers for which the similarity to each of the problem's relationships is maximal. For the example, this involves the calculation of a set of similarity values Θ_i for each answer A_i:

$$\boldsymbol{\Theta_i \leftarrow \{ S(\ H1, H(A_i)\ ), S(\ H2, H(A_i)\ ), S(\ V1, V(A_i)\ ), S(\ V2, V(A_i)\ )\ \}}$$

$$\boldsymbol{\forall\ i,\ 1 \leq i\ \leq\ 8}$$

where S(X,Y) is the Tversky similarity between two sets X and Y, and $H(A_i)$ and $V(A_i)$ denote the relationship formed when the answer image A_i is included in the H() or V() set, respectively.

Reconciling Multiple Analogical Relationships

For each candidate answer, the similarity of each potential analogical relationship is considered as a value upon an axis in a large "relationship space." The dimensionality of this space is determined by the problem at hand. Thus, for a 2x2 Raven's problem, the space is 2-dimensional; for a 3x3 Raven's problem, the space is 4-dimensional, using the relationships as shown above.

A single value for the similarity is desired. To do so, I treat these multidimensional sets as a vector, and determine its length, using a Euclidean distance formula:

$$\mathbf{Si \leftarrow \sqrt{\Sigma\, \Theta_{ij}^{2}} \quad \forall\, i,\, 1 \leq i \leq 8 \text{ and } \forall\, \Theta_{ij} \in \Theta_{i}}$$

Thus, the longer the vector, the more similar; the shorter the vector, the more dissimilar.

Generally, no particular relationship is favored; that is, I do not, as an example, weight more decisively those values found upon the horizontal relationships over those upon the vertical relationships. Giving preferential weighting to a relationship is a straightforward extension to the calculation above, but choosing which relationship to prefer may be non-trivial.

The Fractal Raven Algorithm

My algorithm for solving Raven's problems is itself a slight modification of the Extended Analogy By Recall (ABR*) algorithm. I now present the Fractal Raven algorithm in pseudo-code form. I separate the algorithm into two parts: the preparatory stage and the execution stage.

Given an image P containing a Raven's problem, determine an answer.

PROBLEM SEGMENTATION

By examination, divide P into two images, one containing the matrix and the other containing the possible answers. Further divide the matrix image into an ordered set of either 3 or 8 matrix element images, for 2x2 or 3x3 matrices respectively. Likewise, divide the answer image into an ordered set of its constituent individual answer choices.

Let $M := \{ m_1, m_2, ... \}$ be the set of matrix element images.
Let $C := \{ c_1, c_2, c_3, ... \}$ be the set of individual answer choices.
Let η be an integer denoting the order of the matrix image (either 2 or 3, for 2x2 or 3x3 matrices respectively).

RELATIONSHIP DESIGNATIONS

Let R be a set of relationships, determined by the value of η as follows:
If $\eta = 2$:
   $R \leftarrow \{ H_1, V_1 \}$ where
   $H_1 \leftarrow$ MutualFractal(m_1, m_2)
   $V_1 \leftarrow$ MutualFractal(m_1, m_3)
Else: (because $\eta = 3$)
   $R \leftarrow \{ H_1, H_2, V_1, V_2 \}$ where
   $H_1 \leftarrow$ MutualFractal(m_1, m_2, m_3)
   $H_2 \leftarrow$ MutualFractal(m_4, m_5, m_6)
   $V_1 \leftarrow$ MutualFractal(m_1, m_4, m_7)
   $V_2 \leftarrow$ MutualFractal(m_2, m_5, m_8)

ABSTRACTION LEVEL PREPARATION

Let d be the largest pixel dimension for any image in the set $M \cup C$.

Let $A := \{ a_1, a_2, ... \}$ represent an ordered range of abstraction values where
   $a_1 \leftarrow d$, and $a_i \leftarrow \frac{1}{2} a_{i-1}\ \forall i, 2 \leq i \leq \lfloor \log 2\ d \rfloor$ and $a_i \geq 2$

The values within A constitute the grid values to be used when partitioning the problem's images.

Algorithm 5.1. The Fractal Raven Algorithm, preparatory stage.

The Fractal Raven Algorithm: Preparatory Stage

In the first stage of the Fractal Raven Algorithm, an image containing the entire problem is first segmented into its component images (the matrix of images, and the possible answers). Next, based upon the complexity of the matrix, the algorithm determines the set of relationships to be evaluated. Then, a range of abstraction levels is determined.

As I have implemented it, the abstraction levels are determined to be a partitioning of the given images into gridded sections at a prescribed size and regularity.

Given M, C, R, A, and η as determined in the preparatory stage, find the answer.

PREPARATORY

Let E be a real number which represents the number of standard deviations beyond which a value's answer may be judged as "confident"

Let S(X,Y) be the Tversky similarity metric for sets X and Y

EXECUTION

For each abstraction $a \in A$:

- Re-represent each fractal representation $r \in R$ according to abstraction a
- $S \leftarrow \varnothing$
- For each answer image $c \in C$:
  - If $\eta = 2$:
    $H \leftarrow$ MutualFractal(m_3, c) according to abstraction a
    $V \leftarrow$ MutualFractal(m_2, c) according to abstraction a
    $\Theta \leftarrow \{ S(H_1, H), S(V_1, V) \}$
  - Else: (because $\eta = 3$)
    $H \leftarrow$ MutualFractal(m_7, m_8, c) according to abstraction a
    $V \leftarrow$ MutualFractal(m_3, m_6, c) according to abstraction a
    $\Theta \leftarrow \{ S(H_1, H), S(H_2, H), S(V_1, V), S(V_2, V) \}$
  - Calculate a single similarity metric from vector Θ:
    $t \leftarrow \sqrt{\Sigma\, \theta^2} \quad \forall\, \theta \in \Theta$
    $S \leftarrow S \cup \{ t \}$
- Set $\mu \leftarrow$ mean (S)
- Set $\sigma_\mu \leftarrow$ stdev (S)$/\sqrt{n}$
- Set $D \leftarrow \{ D_1, D_2, D_3, D_4, \dots D_n \}$ where $D_i = (S_i - \mu)/\sigma_\mu$
- Generate the set $Z := \{ Z_i \dots \}$ such that $Z_i \in D$ and $Z_i > E$
- If $|Z| = 1$, return the answer image $C_i \in C$ which corresponds to Z_i
- otherwise there exists ambiguity, and further refinement must occur.

If no answer has been returned, then no answer may be given unambiguously.

Algorithm 5.2. The Fractal Raven Algorithm, execution stage.

The Fractal Ravens Algorithm: Execution Stage

The algorithm concludes by using a variant of the ABR* algorithm to determine the confidence in the answers at each level, stopping when ambiguity is sufficiently resolved. Thus for each level of abstraction, the relationships implied by the kind of Raven's problem (2x2 or 3x3) are re-represented into that partitioning. Then, for each of the candidate images, a potentially analogous relationship is determined for each of the existing relationships and a similarity value calculated. The vector of similarity values is reduced via a simple Euclidean distance formula to a single similarity. The balance of the Fractal Ravens algorithm follows the ABR* algorithm, using the deviation from the mean of these similarities, continues through a variety of levels of abstraction, looking for an unambiguous answer that meets a specified confidence value.

The example, solved.

Table 5.1 shows the results of running the Fractal Ravens algorithm on the example problem, starting at an original gridded partitioning of 200x200 pixels (the maximal pixel dimension of the images), and then refining the partitioning down to a grid of 3x3 pixels. The table gives the mean (μ), standard deviation (σ_{μ}), and number of features (f) for each level of abstraction (grid). The deviation and confidence for each candidate answer are given for each level of abstraction as well. A confidence level of 95% is sought. In the table, I color a cell yellow if it exceeds the desired confidence level, and red if it does so unambiguously for the given grid partitioning.

Table 5.1. Image Deviations and Confidences

image	deviations & confidences						
	0.175 13.84%	-2.035 -95.82%	-1.861 -93.72%	0.698 51.47%	-0.760 -55.29%	-0.610 -45.79%	-1.311 -81.02%
	-0.321 -25.17%	4.166 >99.99%	2.783 99.46%	2.179 97.07%	0.681 50.4%	1.106 73.12%	0.480 36.86%
	6.390 100%	3.484 99.95%	2.930 99.66%	4.487 >99.99%	3.961 99.99%	4.100 >99.99%	4.006 >99.99%
	0.495 37.97%	-3.384 -99.93%	-3.841 -99.99%	-4.848 <-99.99%	-4.958 <-99.99%	-5.454 -100%	-4.620 <-99.99%
	-1.741 -91.84%	-1.678 -90.67%	-2.148 -96.83%	-0.591 -44.56%	-2.825 -99.53%	-1.921 -94.52%	-2.775 -99.45%
	-0.321 -25.17%	1.560 88.12%	2.444 98.55%	-1.361 -82.64%	0.896 62.96%	0.643 47.99%	0.832 59.49%
	-1.741 -91.84%	0.254 20.02%	2.172 97.02%	-1.826 -93.22%	0.668 49.58%	0.213 16.85%	0.570 43.15%
	-2.935 -99.67%	-2.366 -98.20%	-2.479 -98.68%	1.262 79.31%	2.338 98.06%	1.922 94.54%	2.817 99.52%
grid size	200	100	50	25	12	6	3
μ	0.589	0.310	0.432	0.690	0.872	0.915	0.948
σ_μ	0.031	0.019	0.028	0.015	0.007	0.005	0.003
codes	6	24	96	384	1734	6936	26934
features	378	1512	6048	24192	109242	436968	1696842

Discussion of the example results

The deviations presented in table 5.1 appear to suggest that if one starts at the very coarsest level of abstraction, the answer is apparent (image choice 3). Indeed, the confidence in the answer never dips below 99.83%, across all levels of abstraction.

I see evidence that operating with either too sparse a data set (at the coarsest) or with too homogeneous a data set (at the finest) may be problematic. The coarsest abstraction (200 pixel grid size) offers 378 features, whereas the finest abstraction (3 pixel grid size) offers more than 1.5 million features for consideration.

The data in the table continues to suggests the possibility of automatically detecting these boundary situations. The average similarity measurement at the coarsest abstraction is 0.589, but then falls, at the next level of abstraction, to 0.310, only to thereafter generally increase. This constitutes further evidence for an emergent boundary for coarse abstraction.

I suspect that ambiguity exists for ranges of abstraction, only to vanish at some appropriate levels of abstraction, and then reemerges once those levels are surpassed. I see evidence of such behavior in this example, where there exists ambiguity at grid sizes 100, 50, 25, and 12, then the ambiguity vanishes for grid size 6, and then reemerges for grid size 3. This suggests that there are features within the image which are sufficiently discriminatory only at certain levels of abstraction.

Results of Fractal Ravens on the Raven's progressive matrices

I have tested my Fractal Ravens algorithm on all problems associated with the four main variations of the Raven's Progressive Matrices Tests: 60 problems of the Standard Progressive Matrices test, 48 problems of the Advanced Progressive Matrices test, 36 problems of the Coloured Progressive Matrices test, and 60 problems of the SPM Plus test. To my knowledge, this is the first account of any computational model's attempt at the entire Raven's test. In this section, I present my results and discuss my findings.

Inputs used for the test

To create inputs for the Fractal Ravens algorithm, each page from the various Raven test booklets were scanned, and the resulting greyscale images were rotated to roughly correct for page alignment issues. Then, the images were sliced up to create separate image files for each entry in the problem matrix and for each answer choice.

These separate images were the inputs to the technique for each problem. No further image processing or cleanup was performed, despite the presence of numerous pixel-level artifacts introduced by the scanning and minor inter-problem image alignment issues. Additionally, the fractal algorithm attempted to solve each problem independently: no information was carried over from problem to problem, nor from test variant to test variant. The correct answers for the individual problems were provided in an answer key that came with the source material for each test suite.

The code used in conducted these runs is precisely the same code as used in the earlier example. This code is available for download from our lab website. The images scanned, however, are copyrighted and thus are not available for download. However, I believe that the instructions for preparing the images provided above will allow for someone with access to the Ravens materials to reproduce these results.

Levels of abstraction considered and calculations performed

The images associated with each problem had a maximum pixel dimension of between 150 and 250 pixels. Accounting for variation within each test problem, and setting a minimum grid size of 4 pixels, the algorithm therefore calculated five or six levels of abstraction for each problem, using the formula described above for determining maximum grid size and using a strategy of halving the pixel dimension at each successively finer level of abstraction.

At each level of abstraction, the similarity value for each possible answer was calculated, as proscribed by the Fractal Ravens algorithm. Those calculations used the Tversky formula, and set alpha to 1.0 and beta equal to 0.0, conforming to values used in the coincidence model by Bush and Mosteller (1953). From those values, the mean and standard deviation were calculated, and then the deviation and confidence for each answer was determined. Which answers provided a confidence above the chosen level were noted, as well as whether for each abstraction level the answer was unambiguous or ambiguous, and if ambiguous, in what manner. In those cases where ambiguity was found, I explored several different data techniques to assist in the resolution. I describe those techniques after the presentation of the performance.

Assessment of Fractal Ravens performance against human norms

There are three main assessments that can be made following the administration of a Raven test to an individual: the total score, which is given simply as the number of correct answers; an estimate of consistency, which is obtained by comparing the given score distribution to the expected distribution for that particular total score; and the percentile range into which the score falls, for a given age and nationality (Raven et al. 2003). A score is "consistent" if the difference between the actual score and the expected score for any given set is no more than ± 2 (Raven et al. 2003).

The Standard Progressive Matrices test

The Raven's Standard Progressive Matrices test consists of 60 visual analogy problems, organized into five sets of 12 problems each. The problem sets are denoted by the letters A through E. The problems are ordered in approximate degree of difficulty by set, but this increase in difficulty is not uniform.

Performance on the Standard Progressive Matrices test

On the Raven's Standard Progressive Matrices test, the Fractal Ravens algorithm detected the correct answer at a 95% or higher level of confidence on 50 of the 60 problems. The number of problems with detected correct answers per set were 12 for set A, 10 for set B, 11 for set C, 9 for set D, and 8 for set E. Of the 50 problems where the correct answers detected, 38 were determinable by one or more of the ambiguity-resolution strategies. Of the remaining 12 problems noted answers, all but three were ambiguous between two or three particular answers.   Table 5.2 provides a summarization of these results.

Table 5.2. SPM Results

SPM	Detected Correct	Determined by Strategy	Ambiguous between 2 or 3
Total	50	38	9
set A	12	11	1
set B	10	8	1
set C	11	8	3
set D	9	6	2
set E	8	5	3

As shown in Chart 5.1, the score differences for Fractal Ravens on each set were no more than ±1. For a human test-taker, this score distribution generally would indicate that the test results do provide a valid measure of the individual's general intellectual capacity. This score pattern illustrates that the results achieved by the algorithm fall well within typical human norms on the SPM for all sets.

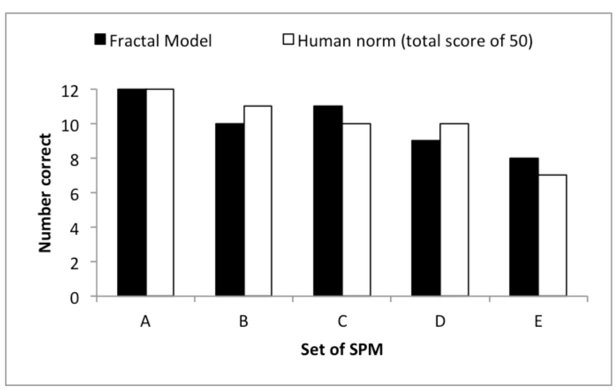


Chart 5.1. Human norms comparison for SPM score of 50.

Using norms from the United States, a total score of 50 corresponds to the 95th percentile for children about 12 years old, the 75th percentile for children around 14 years old, and the 50th percentile for children older than 16 years old (Raven et al. 2003).

The Advanced Progressive Matrices test

The Raven's Advanced Progressive Matrices test consists of 48 visual analogy problems, organized into two sets of 12 and 36 problems, respectively. The problem sets are denoted by the letters A and B. The problems are ordered in approximate degree of difficulty by set, but this increase in difficulty is not uniform.

Performance on the Advanced Progressive Matrices test

On the Raven's Advanced Progressive Matrices test, the Fractal Ravens algorithm detected the correct answer at a 95% or higher level of confidence on 42 of the 48 problems. The number of problems with detected correct answers per set were 10 for set A, and 32 for set B. Of the 42 problems where the correct answers detected, 28 were determinable by one or more of the ambiguity-resolution strategies. Of the remaining 14 problems noted answers, all but four were ambiguous between two or three particular answers. Table 5.3 provides a summarization of the APM results.

Table 5.3. APM Results

APM	Detected Correct	Determined by Strategy	Ambiguous between 2 or 3
Total	42	28	10
set A	10	6	4
set B	32	22	6

The score differences for Fractal Ravens on both APM sets were no more than ±1, indicating consistency and that the results achieved by the algorithm fall well within typical human norms on the APM for both sets. A total score of 42 corresponds to the 95th percentile for adults between 50 and 60 years old, and exceeds the 75th percentile performance for adults of all measured ages (Raven et al. 2003).

The Coloured Progressive Matrices test

The Raven's Coloured Progressive Matrices test consists of 36 visual analogy problems, organized into three sets of 12 problems. The problem sets are denoted by the letters A, AB, and B. The problems are ordered in approximate degree of difficulty by set, but this increase in difficulty is not uniform.

Performance on the Coloured Progressive Matrices test

On the Raven's Coloured Progressive Matrices test, the Fractal Ravens algorithm detected the correct answer at a 95% or higher level of confidence on 30 of the 36 problems. The number of problems with detected correct answers per set were 12 for set A, 11 for set AB, and 7 for set B. Of the 30 problems where the correct answers detected, 24 were determinable by one or more of the ambiguity-resolution strategies. Of the remaining 6 problems noted answers, all were ambiguous between two or three particular answers. Table 5.4 provides a summarization of these results.

Table 5.4. CPM Results

CPM	Detected Correct	Determined by Strategy	Ambiguous between 2 or 3
Total	30	24	6
set A	12	11	1
set AB	11	8	3
set B	7	5	2

As shown in Chart 5.2, the score differences for Fractal Ravens on each set were no more than ±2, indicating consistency. This score pattern also illustrates that the results achieved by the algorithm fall well within typical human norms on the CPM for all sets.

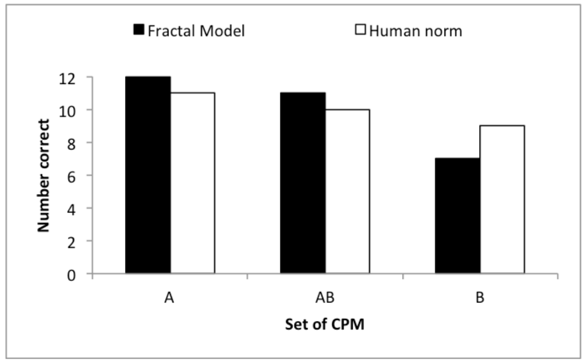


Chart 5.2. Human norms comparison for CPM score of 30.

Using the United States norms, a total score of 30 on the CPM test corresponds to the 95th percentile for children about 7 years old, the 75th percentile for children about 9 years old, and the 50th percentile for children about 11 years old (Raven et al. 2003).

The SPM Plus test

The Raven's Standard Progressive Matrices Plus test consists of 60 visual analogy problems, organized into five sets of 12 problems. The problem sets are denoted by the letters A, B, C, D, and E. The problems are ordered in approximate degree of difficulty by set, but this increase in difficulty is not uniform.

Performance on the SPM Plus test

On the Raven's Standard Progressive Matrices Plus test, the Fractal Ravens algorithm detected the correct answer at a 95% or higher level of confidence on 50 of the 60 problems. The number of problems with detected correct answers per set were 10 for set A, 9 for set B, 9 for set C, 11 for set D, and 11 for set E. Of the 50 problems where the correct answers detected, 39 were determinable by one or more of the ambiguity-resolution strategies. Of the remaining 11 problems noted answers, all but one were ambiguous between two or three particular answers. Table 5.5 provides a summarization of these results.

Table 5.5. SPM Plus Results

SPM Plus	Detected Correct	Determined by Strategy	Ambiguous between 2 or 3
Total	50	39	10
set A	10	9	1
set B	9	8	1
set C	9	5	4
set D	11	8	2
set E	11	9	2

Comparison to other computational models

While this is the first published account of a computational model's attempt at the entire suite of Raven's tests, there are other computational models which have been used on some or all of certain tests. For those accounts which report scores, I compared their results with those achieved by the Fractal Raven algorithm. Table 5.6 documents the performance of the Fractal Ravens algorithm against those contemporaries.

Table 5.6 Comparing Fractal Ravens to Other Models

	SPM Results							APM Results			
model	Total	# att.	A	B	C	D	E	Total	# att.	A	B
Carpenter et al. "FairRaven"								23	34	7	16
Carpenter et al. "BetterRaven"								32	34	7	25
Lovett et al. (2007)	22	24	-	12	10	-	-				
Lovett et al. (2010)	44	48	-	unreported							
Cirillo & Ström	28	36	-	-	8	10	10				
Kunda et al. Affine	50	60	11	12	10	8	9	18	48	5	13
Fractal Ravens	**50**	**60**	**12**	**10**	**11**	**9**	**8**	**42**	**48**	**10**	**32**

Carpenter et al. (1990) report results of running two versions of their algorithm (FairRaven and BetterRaven) against a subset of the APM problems. The subset of problems chosen by Carpenter et al. reflect those whose rules and representations were deemed as inferable by their production rule based system (Carpenter et al. 1990).

Lovett et al. (2007, 2010) report results from their computational model's approach to the Raven's SPM test. In each account, only a portion of the test was attempted, but Lovett et al. project an overall score based on the performance of the attempted sections. The latest published account by Lovett et al. (2010) reports a score of 44 out of 48 attempted problems from sets B through E of the SPM test, but does not offer a breakdown of this score by problem set. Lovett et al. (2010) project a score of 56 for the entire test, based on human normative data indicating a probable score of 12 on set A given their model's performance on the attempted sets.

Cirillo and Ström (2010) report that their system was tested against Sets C through E of the SPM and solved 8, 10, and 10 problems, respectively. Though unattempted, they predict that their system would score 19 on the APM (a prediction of 7 on set A, and 12 on set B).

Kunda et al. (2011, 2012) report the results of running their Affine algorithm against all of the problems on both the SPM and the APM tests, with a detailed breakdown of scoring per test. They report a score of 50 for the SPM test, and a score of 18 on the APM test.

Specific comparison of Fractal Ravens vs Kunda et al. Affine

The agreement of scores between the Fractal Ravens algorithm and the Kunda et al. Affine algorithm on the SPM warrant further inspection and remarks. Kunda et al. (2012) inspect each row and column of a Raven problem, comparing pixels between images under both a series of similitude transformations (indeed, they employ the same eight similitude transformations used by the fractal encoding process) and other pixel transformations. Once a candidate transformation has been selected, then the transformation is applied to the images in the final row and column of the problem, generating a prediction image. This prediction image is then compared against the candidate images by calculating a similarity score based on pixel correlation. The candidate image with the maximum similarity score is selected as the answer.

Although Kunda et al. (2012) do not report the ambiguity of their results, I received their similarity data for the SPM, APM, and CPM test results via private communication, and made the calculation using the techniques described above for the ABR* algorithm. Table 5.7 directly compares the specific results of the Kunda et al. Affine algorithm and the Fractal Ravens, color coded to indicate ambiguity.

Table 5.7. Comparison of SPM Results between Affine and Fractal Ravens

Table 5.7 shows the results of both algorithms at a 95% level of confidence. While both algorithms answer 50 of the 60 problems, the degree of ambiguity differs substantially. The Fractal Ravens algorithm has ambiguous results on 12 problems, while the Affine algorithm is ambiguous on 31 problems. This discrepancy can be attributed at least in part to the single level of abstraction at which the Affine algorithm operated. In contrast, the Fractal Ravens algorithm examined five or six levels of abstraction for each problem.

A	Fractal	Affine
#	12	11
1	*correct*	*correct*
2	*correct*	*correct*
3	*correct*	*ambiguous*
4	*correct*	*correct*
5	*correct*	*correct*
6	*correct*	*correct*
7	*ambiguous*	*ambiguous*
8	*correct*	*correct*
9	*correct*	*correct*
10	*correct*	*ambiguous*
11	*correct*	
12	*correct*	*ambiguous*

B	Fractal	Affine
#	10	12
1	*correct*	*correct*
2	*correct*	*correct*
3	*ambiguous*	*correct*
4	*correct*	*correct*
5	*correct*	*correct*
6	*correct*	*correct*
7	*ambiguous*	*ambiguous*
8	*correct*	*correct*
9		*ambiguous*
10		*ambiguous*
11	*correct*	*correct*
12	*correct*	*correct*

C	Fractal	Affine
#	11	10
1	*correct*	*ambiguous*
2	*correct*	
3		*ambiguous*
4	*correct*	*ambiguous*
5	*ambiguous*	*ambiguous*
6	*correct*	*correct*
7	*correct*	*ambiguous*
8	*correct*	*ambiguous*
9	*correct*	
10	*ambiguous*	*ambiguous*
11	*ambiguous*	*ambiguous*
12	*correct*	*ambiguous*

at more coarser and more fine levels of abstraction, other answers would have been selected. Thus, … gorith… , it cannot be selected unamb…

D	Fractal	Affine
#	9	8
1	*correct*	*ambiguous*
2	*correct*	*ambiguous*
3	*correct*	*ambiguous*
4	*ambiguous*	*ambiguous*
5	*correct*	*ambiguous*
6	*correct*	
7	*ambiguous*	
8		
9	*ambiguous*	*ambiguous*
10		*ambiguous*
11		*ambiguous*
12	*correct*	

E	Fractal	Affine
#	8	9
1		*ambiguous*
2	*correct*	*ambiguous*
3		*ambiguous*
4	*correct*	*correct*
5	*correct*	*correct*
6	*ambiguous*	*ambiguous*
7	*ambiguous*	
8	*correct*	*ambiguous*
9	*ambiguous*	*ambiguous*
10		
11	*correct*	*ambiguous*
12		

Each al… rect a… they fail in common on onl… and … pancy for the remaining 14 p… gorie… Affine incorrect (A11, … ctal R… ncorrect (D7 and E7); and Fr… fine a… 10, D11, E1, and E3). There… he Af… while the Fractal Ravens …

With re… liscre… rrect / Affine ambiguous) in which the Affine algorithm outperformed the Fractal Ravens algorithm, three of the problems (C3, E1, and E3) involve the addition of two of the three horizontal or vertical images. The Affine algorithm employs a specific pixel addition transformation as well as the eight similitude transformations, and it is likely that this accounts for the performance distinction on these problems. In the remaining four problems, the Fractal

Ravens algorithm fails to note the answer unambiguous at the considered levels of abstraction. Three problems (B9, B10 and D10) have the answer noted as among the best three answers at certain levels, but in each instance noted below the 95% confidence level. The remaining problem (D11) has the answer noted as an ambiguous possibility, but prefers instead to choose an answer which has the correct aspect (closed figure) and shape (wave), but is truncated instead of full.

Thus, while the scores of the two algorithms are in close agreement, the differences in the algorithms' approaches and the representations used lead to specific distinctions in the individual answers on the SPM test. As shown, the Fractal Ravens algorithm, by dint of its use of several levels of abstraction, is substantially less ambiguous than the Affine algorithm.

Choice of psychological model

As noted earlier, the Tversky formula for featural similarity offers the ability to vary the similarity metric through the selection of weights for common or unique features. In my assessment, I looked at the Fractal Ravens algorithm's performance on all of the Raven tests using three such psychological models of similarity: the Gregson-Sjoberg model, the Eisler-Ekman model, and the Bush-Mosteller model.

The Gregson-Sjoberg model establishes $\alpha = \beta = 1.0$ This yields a similarity metric which is the Jaccard similarity, a balanced approach which favors neither transformation:

$$\mathbf{S(T,T') = Ƒ(T \cap T') / Ƒ(T \cup T')}$$

Tversky (1977) points out that the Eisler-Ekman model, setting $\alpha = \beta = 0.5$, yields a a similarity metric of the form:

$$\mathbf{S(T,T') = 2Ƒ(T \cap T') / (Ƒ(T) + Ƒ(T'))}$$

A slightly different formulation for Eisler-Ekman is given by Junge (1977):

$$\mathbf{S(T,T') = 2q / (1 - q) \quad where\ q = RT/RT'}$$

where the values RT and RT' are the responses to stimuli T and T'. Junge's reformulation using the response ratio introduces an asymmetry (through the choice of which response is to be judged as the denominator).

Of the models considered, only the Bush-Mosteller model, which sets $\alpha = 1.0$ and $\beta = 0$, offers a strictly Tversky-formulated asymmetric view of the transformations. In doing so, this model introduces the notion of directional salience (Santini & Jain, 1999). This asymmetry creates a violation of the strictly geometric distance axioms associated with a distance metric. Another interpretation is that using the Bush-Mosteller model,

$$\mathbf{S(T,T') > S(T',T) \text{ whenever } Ƒ(T) < Ƒ(T')}$$

This implies a relationship between the asymmetry of the model and the homogeneity of the features being used to discrimination T and T'.

In my experiments, I found that the choice of model had little or no effect on the results obtained on the APM and CPM tests. On the SPM and the SPMPlus tests, the outcomes did change, but only very slightly. The Bush-Mosteller model generated the highest score on the SPM (a score of 50), while the Gregson-Sjoberg model generated the lowest score of 48. I conclude that the selection of the psychological model for similarity distance has little effect on the general outcome of the Fractal Ravens, though testing other models and similarity metric calculations would be a good exercise for future research.

I speculate that the lack of outcome differential when using the Bush-Mosteller model in my results is due to the juxtaposition of that model's implicit directional salience with the mutual fractal representation. I will offer further speculation on the relevance of this model a bit further ahead in this discussion.

Performance at varying levels of confidence

I also ran the Fractal Ravens algorithm against the Ravens test suite using a variety of levels of confidence. Chart 5.3 provides the details of these test runs.

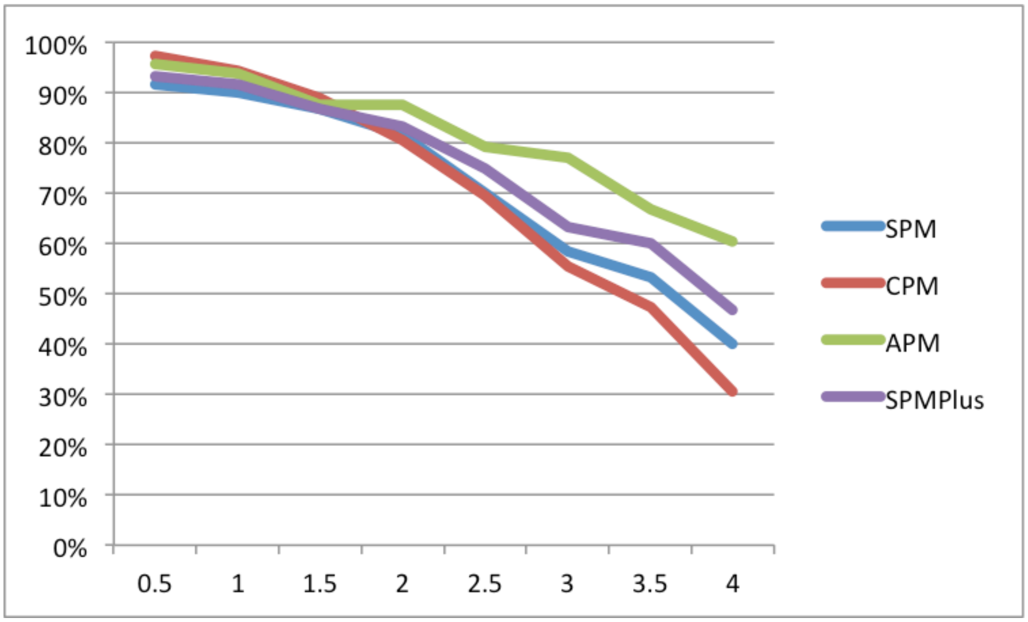


Chart 5.3. Percentage of Correct Scores, Performance per sigma

As the confidence level, expressed here in terms of the deviation, increased from 38% confidence (at 0.5-sigma) to 99.99% confidence (at 4-sigma), the test performance decreased. Note that at 95% confidence (about 2-sigma), for all tests, the scores are at or above 80%, and there is a sharp falloff in certain tests (the CPM in particular) for 3-sigma and beyond. Even so, the SPM and SPMPlus, while different in content but similar in composition, exhibit performance curves which resemble one another.

As confidence increases, the agreement in scores between those noted correct and those discernible as correct via some strategy also increases. I believe my computational evidence suggests that increasing confidence decreases ambiguity, and thus provides sufficient data for the various strategies to determine an answer. That, in turn, leads to the convergence in score agreement as confidence increases.

CHAPTER 6

FRACTALS AND MILLER ANALOGIES

In this chapter, I describe the use of a derivation of the Extended Analogy By Recall (ABR*) algorithm on a classic set of geometric analogy problems, those first used by Evans in his 1964 study (Evans, 1964). This chapter develops and illustrates the Fractal Miller algorithm, and provides a comparison of its performance against contemporary studies and human behavior on the Evans suite of problems.

Miller Analogies Test

The Miller Analogies Test (MAT) is a high-level mental ability test which requires the solution of problems presented as analogies (Meagher, 2006; Pearson 2011). The test is used by a large number of graduate studies programs as one of several criteria for admission, as the abilities to recognize and to construct analogies are thought to be key indicators of constructing explanations and building arguments, and represents a fundamental way in which understanding is formed and communicated (Gentner, Holyoak, & Kokinov, 2001; Holyoak & Thagard, 1996). Psychologists also suggest that the format of the Miller Analogy Test represents an efficient and effective way to sample reasoning processes and to measure verbal reasoning, inferential ability, and analytical intelligence (Kuncel, Hezlett, & Ones, 2004; Lohman, 2004; Sternberg, 1977, 1985, 1988).

MAT Analogies

The problems on the MAT are entirely verbal, and are referred to as MAT analogies. MAT analogies have the general form "A : B :: C : ____" with four possible answer choices given. To solve a MAT analogy, one must recognize some relationship between two of the given terms, and then look for that same relationship between the third given term and one of the answer choices.   This yields two possible interpretations for "A : B :: C : ___":

A is to B as C is to (one of the answers)

A is to C as B is to (one of the answers)

The Miller Analogy Test expressly precludes the remaining interpretation (Pearson 2011):

B is to C as A is to (one of the answers)

Here are some examples of MAT analogies:

Plane : Air :: Car : _____

(a . motorcycle, b . engine, c . land, d . atmosphere)

Induction : _____ :: Soldier : Priest

(a . confirmation, b . graduation, c . ordination, d . resistance)

In the first example, the sought-for relationship is "travels by": a plane "travels by" air. Thus, one can consider in turn which of the possible answers would best satisfy the statement: car "travels by" ____. In this example, the answer is "c. land."   In the second example, the sought-for relationship is "ceremony for becoming": a soldier becomes one by being inducted. The answer to the second example, then, is "c. ordination": a priest becomes one by being ordained.

Note also that the second example is given in a format which is different but semantically equivalent: C : ____ :: A : B. Indeed, on the Miller Analogy Test, the problem may be posed in any of these formats:

A : B :: C : _____

A : B :: _____ : C

C : _____ :: A : B

_____ : C :: A : B

Regardless of the manner in which the missing term is presented, the challenge of solving a MAT analogy remains the same: there will exist one relationship which best describes either the pair A and B or the pair A and C, and therefore one sought-for answer.

Solving a MAT Analogy

Schematically, if one treats the original analogy as A : B :: C : X, then there are two candidate relationships R_1 and R_2 which can be expressed in a functional notation:

$$\mathbf{R_1(A,B) :: R_1(C,X)}$$

$$\mathbf{R_2(A,C) :: R_2(B,X)}$$

The challenge with solving a MAT analogy, then, is to determine which of these relationships R_1 or R_2 is the sought-for relationship. This determination will be informed by all of the given terms (A,B,C), as well as the four potential answers to be substituted in for X.

Geometric variations of the Miller Analogy Test and AI

In 1964, Evans published a paper entitled "A heuristic program to solve Geometric Analogy Problems" (Evans, 1964). In the paper, he describes his efforts to address a ‘wide class of intelligence test problems of the “geometric-analogy” type (“figure A is to figure B as figure C is to which of the following figures?”).’ What Evans proposed, and produced, was the first such program to tackle geometric variations of the Miller Analogy test.

Evans used a canonical format for his problems: A : B :: C : (a, b, c, d, e), with three given images (A, B, C), and five potential answers (a,b,c,d,e). An example problem from those Evans addressed is shown in Figure 6.1.

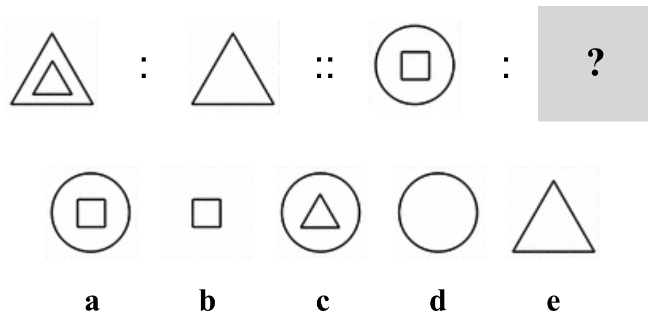


Figure 6.1. An Evans analogy problem

Evans (1964) notes that his approach does not concern itself with the original capture of the figural information, and instead presumes the capture process results in a list-structure representation, comprised of geometric descriptions. For example, (DOT (X . Y)) would be inferred to mean that there is a DOT at coordinates (X,Y), and (SCC ((X_1 . Y_1) 0.0 (X_2 . Y_2) 0.0 (X_3 . Y_3) 0.0 (X_1 . Y_1))) would be inferred as a “simple closed curve” describing a triangle with vertices at (X_1,Y_1), (X_2,Y_2), and (X_3,Y_3). Other descriptions included ways to denote spatial relationships between pairs of objects (e.g. (INSIDE A B) meant that object A was wholly contained inside of object B). Each of the given objects in a geometric MAT problem and all of the possible answers were represented in this manner.

With this representation of the problem, Evans made another simplifying decision and restricted his work to being an interpretation of the R_1 relationship of the more general MAT problems; that is, the form A : B :: C : X is always interpreted as $R_1(A,B)$ :: $R_1(C,X)$, removing the problem of determining which of the two relationships should be sought. This in turn provided two-part strategy for Evan's solver. First, the solver determined all of the possible ways in which the spatial relationships of A and B can be matched, and from those matchings established one or more rules for transforming A into B (thereby determining $R_1(A,B)$). Next, a candidate set of rules is determined for transforming C into each of the potential answers, using the same technique. Then, each of those rules are compared against the set of $R_1(A,B)$ rules, and the similarity between each is noted (in terms of how much or little of the original rule appears in the candidate rule, and the manner in which the variables are used). Finally, the rules which are found to resemble each other are applied to the objects A and C, to determine which information is preserved in the construction of B from A and the construction of the candidate answer from C (Evans, 1964).

Evans (1964) points out that although there may be many such rules determined by the first part of the solver, he coded the solver to seek the "best" or "strongest" rule, one that results in the most descriptive answer or is the least alteration of the rule (Evans permits for the production of an answer which uses some but not necessarily all of the elements of a rule). As Evans notes, if a single rule meets this criteria, it is chosen as the analogy, and the potential answer from which it was generated is selected as the answer to the problem (Evans, 1964). If a tie results, the method is deemed to have failed to produce an analogy.

Evans' aim was to develop a method by which a program could form a "theory" – the $R_1(A,B)$ rule – on the basis of the evidence present in the descriptions of A and B, and then generalize the theory as required (through rule modification) to fit further evidence (in the form of C). Once generalized, the program would use the theory to make a prediction from that evidence, and test that prediction via comparison of the output to the potential answers, iteratively modifying the rule and retesting its predictions until a singular answer was determined or no further modifications could be formed.

Contemporary approaches to Miller Geometric Analogies

While there are several published accounts of models and systems which address geometric analogies (Bohan & O'Donoghue 2000; Ragni et al., 2007; Schwering et al., 2007), the largest body of contemporary work on address the specific challenges of the Evans set of geometric MAT problems comes from the work of Kenneth Forbus and his colleagues. Several of the publications report on the problem of constructing input from sketches (Forbus et al., 2001; Forbus et al., 2008). Two papers in particular, however, specifically address geometric MAT problems in details.

In Tomai et al. (2004), the authors develop and defend the notion that qualitative visual structure combined with analogical processing can produce human-like results, using the domain of geometric MAT problems. The paper illustrates the general purpose nature of the Structure-Mapping Engine (SME). With regard to Evans, however, Tomai et al. demonstrate that processing only differences (as opposed to similarities) in the second stage of their model leads to sufficient results.

In Lovett et al. (2009), the authors further their argument that second-order analogies over differences computed via analogies between images are sufficient. They make an extensive comparison between the results of their model's output and that of a behavioral experiment performed on humans taking the Evans problem suite.

The Fractal Miller algorithm

In the course of my research, I developed the Fractal Ravens algorithm as presented in an earlier chapter. Since the mechanisms for deriving an answer based upon multiple relationships were already available through that algorithm, the derivation of the Fractal Miller algorithm from the Fractal Ravens algorithm was straightforward.

An example

Let us use the example problem from Evans' research as the basis for describing the algorithm. Recall that in the problem, there is exactly one relationship which must be considered (as opposed to the several that must be simultaneously considered when addressing a Raven's problem).   Let us denote this left-side relationship as R, and the right-side relationship as R', as shown in Figure 6.2.

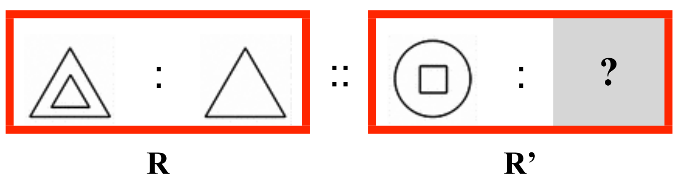


Figure 6.2. the R and R' relationships

To solve a Miller's problem, one must construct a set of similarity values Θ_i for each of the five potential answers X_i:

$$\mathbf{S_i \leftarrow S(\ R,\ R'(X_i)\ )\quad \forall\ i,\ 1 \leq i\ \leq\ 5}$$

where S(X,Y) is the Tversky similarity between two sets X and Y, and $R'(X_i)$ denotes the relationship formed when the answer image X_i is included in the R'() set.

The algorithm, presented

My algorithm for solving geometric MAT problems, like the Fractal Ravens algorithm, is itself a slight modification of the Extended Analogy By Recall (ABR*) algorithm. Here is the algorithm in pseudo-code form. I separate the algorithm into two parts: the preparatory stage and the execution stage.

> Given a geometric Miller's Analogy Test problem, determine an answer.
>
> **PROBLEM SEGMENTATION**
>
> By examination, divide the problem into two segments, one containing the matrix of givens and the other containing the possible answers. Further divide the matrix of givens into an ordered set of 3 images. Likewise, divide the answer segment into an ordered set of its constituent individual answer choices.
>
> Let $M := \{ m_1, m_2, m_3 \}$ be the set of matrix element images.
> Let $C := \{ c_1, c_2, c_3, \ldots \}$ be the set of individual answer choices.
>
> **RELATIONSHIP DESIGNATIONS**
>
> Let R be a relationship, determined as follows:
> $R \leftarrow \text{MutualFractal}(m_1, m_2)$
>
> **ABSTRACTION LEVEL PREPARATION**
>
> Let d be the largest pixel dimension for any image in the set $M \cup C$.
>
> Let $A := \{ a_1, a_2, \ldots \}$ represent an ordered range of abstraction values where
> $a_1 \leftarrow d$, and $a_i \leftarrow \frac{1}{2} a_{i-1}$ $\forall\ i, 2 \leq i \leq \lfloor \log2\ d \rfloor$ and $a_i \geq 2$
>
> The values within A constitute the grid values to be used when partitioning the problem's images.

Algorithm 6.1. The Fractal Miller Algorithm, preparatory stage.

The Fractal Miller Algorithm: Preparatory Stage

In the first stage of our Fractal Miller Algorithm, a geometric Miller's Analogy Test problem is first segmented into its component images (the matrix of the given images, and the collection of images of possible answers). Next, the algorithm determines the relationship between the first two given images, expressed as a mutual fractal representation. Then, a range of abstraction levels is determined.

As I have implemented it, the abstraction levels are determined to be a partitioning of the given images into gridded sections at a prescribed size and regularity.

Given M, C, R, A, and η as determined in the preparatory stage, find the answer.

PREPARATORY

Let E be a real number which represents the number of standard deviations beyond which a value's answer may be judged as "confident"

Let S(X,Y) be the Tversky similarity metric for sets X and Y

EXECUTION

For each abstraction $a \in A$:

- Re-represent each fractal representation $r \in R$ according to abstraction a
- $S \leftarrow \varnothing$
- For each answer image $c \in C$:
  $R' \leftarrow$ MutualFractal(m_3, c) according to abstraction a
  $S \leftarrow S \cup \{ S(R, R') \}$
- Set $\mu \leftarrow$ mean (S)
- Set $\sigma_\mu \leftarrow$ stdev (S)/$\sqrt{n}$
- Set $D \leftarrow \{ D_1, D_2, D_3, D_4, \ldots D_n \}$ where $D_i = (S_i-\mu)/\sigma_\mu$
- Generate the set $Z := \{ Z_i \ldots \}$ such that $Z_i \in D$ and $Z_i > E$
- If $|Z| = 1$, return the answer image $C_i \in C$ which corresponds to Z_i
- otherwise there exists ambiguity, and further refinement must occur.

If no answer has been returned, then no answer may be given unambiguously.

Algorithm 6.2. The Fractal Miller Algorithm, execution stage.

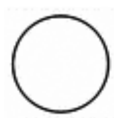
The Fractal Miller Algorithm: Execution Stage

The algorithm concludes by using a variant of the ABR* algorithm to determine the confidence in the answers at each level, stopping when ambiguity is sufficiently resolved. Thus for each level of abstraction, the relationship R is re-represented into that partitioning. Then, for each of the candidate images, a potentially analogous relationship is determined and a similarity value is calculated. The balance of the Fractal Miller algorithm follows the ABR* algorithm, using the deviation from the mean of these similarities, continues through a variety of levels of abstraction, looking for an unambiguous answer that meets a specified confidence value.

The example, solved.

Table 6.1 shows the results of running the Fractal Miller algorithm on the example problem, starting at an original gridded partitioning of 59x59 pixels (the maximal pixel dimension of the images), and then refining the partitioning down to a grid of 7x7 pixels. The table gives the mean (μ), standard deviation (σμ), and number of features (f) for each level of abstraction (grid). The deviation and confidence for each candidate answer are given for each level of abstraction as well. A confidence level of 95% is sought. In the table, I color a cell yellow if it exceeds the desired confidence level, and red if it does so unambiguously for the given grid partitioning.

Table 6.1. Image Deviations and Confidences

image	**deviations & confidences**			
	1.0 68.27%	3.034 99.76%	3.079 99.76%	2.099 96.42%
	1.0 68.27%	1.687 90.85%	1.604 89.13%	1.329 81.61%
	1.0 68.27%	1.005 68.52%	1.069 71.51%	0.886 62.43%
	4.0 99.94%	1.723 91.52%	1.438 84.96%	0.944 65.47%
	1.0 68.27%	1.992 95.37%	2.176 97.04%	3.486 99.95%
grid size	59	29	14	7
μ	0.2889	0.1644	0.1382	0.2033
σ_μ	0.0508	0.0197	0.0069	0.0020
codes	2	18	50	162
features	126	1134	3150	10206

Discussion of the example results

As with the Fractal Ravens example results, here the deviations presented in table 6.1 appear to suggest that if the algorithm starts at the very coarsest level of abstraction, the answer is apparent (image choice 4). The confidence in that answer tapers off as the level of abstraction becomes finer.

Again, as with the Fractal Ravens results, I see evidence that operating with either too sparse a data set (at the coarsest) or with too homogeneous a data set (at the finest) may be problematic. The coarsest abstraction (59 pixel grid size) offers 126 features, whereas the finest abstraction (7 pixel grid size) offers more than 10,000 features for consideration.

The data in the table continues to suggests the possibility of automatically detecting these boundary situations. The average similarity measurement at the coarsest abstraction is 0.2889, and falls steadily at finer levels of abstraction, to sharply increase at the finest level. Unlike the Fractal Ravens example, I believe this provides evidence for for an emergent boundary for finer abstraction.

I note also that the only level for which the answer is unambiguous is the most coarse level of abstraction, and that all other tested levels offer ambiguity. To me, this suggests that while there may be a sufficient level of abstraction (even if it is coarse) at which an unambiguous answer may be obtained, perhaps the fact that there is but only analogical relationship at play in a geometric MAT problem, as opposed to the multiple analogical relationships present in a Raven's problem, implies that the additional constraints of multiple relationships may serve to increase the confidence in the answer.

Running the algorithm against the Evans problems

I tested the Fractal Miller algorithm on all of the problems originally used by Evans in his 1964 paper. I now present the results of that experiment, and a discussion of those results.

Inputs used for the test

To create inputs for this experiment, I used the same problems Evans used, but as illustrated in the appendix of Lovett et al. (2009). A screen capture was made of each page of the appendix, and segmented into individual problem images. Then, each problem image was sliced up to create separate image files for each of the given items in the matrix and for each answer choice. These separate images were the inputs to the technique for each problem. No further image processing or cleanup was performed, despite the presence of several pixel-level artifacts introduced by capture and compression. Additionally, the fractal algorithm attempted to solve each problem independently: no information was carried over from problem to problem, nor from test variant to test variant.

The code used in conducted these runs is precisely the same code as used in the earlier example. This code is available for download from our lab website, along with the images themselves.

Levels of abstraction considered and calculations performed

The images associated with each geometric MAT problem had a maximum pixel dimension of 59 pixels. Accounting for variation within each test problem, and setting a minimum grid size of 7 pixels, the algorithm therefore calculated four levels of abstraction for each problem, using the formula described above for determining maximum grid size and using a strategy of halving the pixel dimension at each successively finer level of abstraction.

At each level of abstraction, the similarity value for each possible answer was calculated, as proscribed by the Fractal Miller algorithm. Those calculations used the Tversky formula, with alpha set to 1.0 and beta equal to 0.0, conforming to values used in the coincidence model by Bush and Mosteller (1953). From those values, the algorithm calculated the mean and standard deviation, and then calculated the deviation and confidence for each answer. Which answers provided a confidence above our chosen level were noted, and whether for each abstraction level the answer was unambiguous or ambiguous, and if ambiguous, in what manner. In those cases where ambiguity was found, I explored several different data techniques to assist in the resolution, the same techniques which I described in the earlier chapter on Fractal Ravens results.

Performance on the Evans suite of geometric MAT problems

On the Evans suite of geometric MAT problems, the Fractal Miller algorithm detected the correct answer at a 95% or higher level of confidence on 13 of the 20 problems. Of the 13 problems where the correct answers detected, 11 were determinable by one or more of our ambiguity-resolution strategies. Of the remaining two problems noted as answers, both were ambiguous between two particular answers.

Comparison to other computational models

This certainly is not the first published account of a computational model's attempt at the Evans suite. For those accounts which explicitly report scores (e.g. Tomai et al., 2004), I compared their results with those achieved by the Fractal Miller algorithm. I also compared the results against human performance data on the problems as given in an experiment detailed in Lovett et al. (2009). Table 6.2 documents the performance.

Table 6.2. Comparison to Other Methods

problem	FractalMiller	Tomai et al. 2004	Human
1	*ambiguous*	*yes*	*yes*
2	*yes*	*yes*	*yes*
3	*yes*	*yes*	*yes*
4	*yes*	*yes*	*ambiguous*
5		*yes*	*yes*
6		*yes*	*yes*
7	*yes*	*yes*	*yes*
8	*yes*	*yes*	*yes*
9		*yes*	*yes*
10	*yes*		*ambiguous*
11	*yes*	*yes*	*yes*
12	*yes*		*yes*
13		*ambiguous*	*ambiguous*

14		*yes*	*ambiguous*
15	*yes*		*yes*
16		*yes*	*yes*
17			*ambiguous*
18	*yes*	*yes*	*ambiguous*
19	*yes*	*yes*	*ambiguous*
20	*ambiguous*	*yes*	*yes*
Total	13 (2 ambiguous)	15 (1 ambiguous)	20 (7 ambiguous)

Tomai et al. (2004) report the results of a version of their algorithm. They specifically suggest that their efforts are not to improve upon the results of Evans' original program, but to validate the ability of their general purpose system, SME, to produce human-like analogy judgments. The representation of the problems are created automatically using sKEA, a sketching knowledge entry associate program. These representations are in turn fed into SME, an implementation of Gentner's structure mapping theory (Gentner 1983). Tomai et al. report that a number of the problems (12-20) were run through only stage 2 of their system due to limitations in recognition. They also note that the program did not perform axial symmetry, lacked the ability to decompose glyphs, lacked a hierarchical awareness in positional relationships, and did not have the ability to reinterpret the example to attempt another solution. They note that these deficiencies were present in some of the correct answers, but other factors were able to deduce a correct answer.

In Lovett et al. (2009) the SME program is again employed to solve the Evans set, using the output of CogSketch as input. In addition to the two-stage process of SME, an additional model component, the executive, is presented as a means to evaluate whether an answer is a sufficient answer (e.g. all the facts determined in the relationships between the givens and the candidate answer align). Additional modes of mapping for the first-stage area also introduced, but their usage is strongly constrained. The results reported in Lovett et al. (2009) concern themselves more with the discussion of their correlation with human-preferred answers rather than providing an explicit account and detailed discussion of their computational models. They do note that their model chooses the human-preferred answer for each of the 20 problems. Thus, their results can be construed as identical to that given in the human column in the table above.

It is also important to note that for the problems in which the human answer is given as "ambiguous" this means that I have interpreted any problem in which less than 95% of the participants preferred that any one particular answer as ambiguous. This decision I made subjectively, but motivated by the desire to mirror the 95% confidence level used in the Fractal Miller algorithm.

Bohan and O'Donoghue (2000) discuss Evans ANALOGY in the context of presenting their argument for adding attribute matching to Gentner's theory (which calls for only relational predicate mapping). However, they offer no evidence of having attempted to solve any of the Evans problems.

Ragni, Schleipen, and Steffenhagen (2007) similarly discuss Evans ANALOGY briefly, but only in the service of contrasting against their SRM model, and its focus and grid approach.

Schwering et al. (2007) offer an approach for solving geometric analogy problems using only a single analogical mapping stage, based on hand-coded representations and Gestalt grouping principles. They offer no evidence of having attempted to solve any of the Evans problems using their Heuristic-Driven-Theory-Projection (HDTP) system.

CHAPTER 7

FRACTALS AND VISUAL ODDITY

This chapter will discuss visual oddity, and a class of problems from visual analogy in which similarity calculations are used to derive an answer. This chapter develops a cognitively-inspired computational model to address that class of problems, and introduces the concept of distributed similarity.

Oddity and Novelty

Let us consider a problem of visual oddity. Suppose one is presented with a group of objects, and are to determine, without further instruction, which one of the objects does not belong with the others. Figure 7.1 shows an example of such a problem.

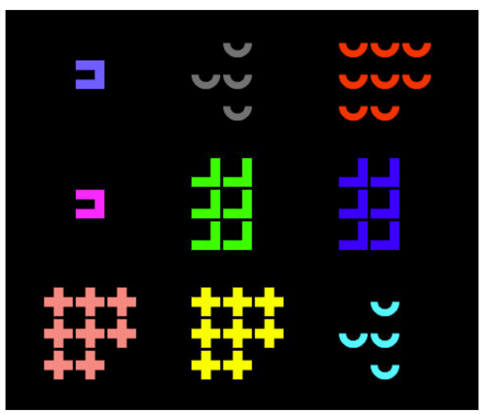


Figure 7.1. Visual Oddity example

Note that although the problem presents the objects in a matrix fashion, the presentation itself is meaningless. The problem of selecting which one does not belong would remain regardless of arrangement.

This problem can be interpreted as a classification problem, where one of the objects is classified into one category (ODD), and the rest are classified into another category (TYPICAL). For example, if there are n objects, and all of the objects O_i are in some set M initially, one can presume that there exists two mutually exclusive sets ODD and TYPICAL:

$$\mathbf{M = \cup\, O_i \quad \forall\, i,\ \ 0 < i \leq n}$$

$$\mathbf{\exists\ ODD, TYPICAL \rightarrow M = ODD \cup TYPICAL}$$

$$\mathbf{and\ ODD \cap TYPICAL = \varnothing}$$

One would seek some function S() which serve to score each object, and then given the score be able to assign the object into one of the two sets. The value of the function would act as a threshold T:

$$\mathbf{O_i \in TYPICAL\ if\ S(O_i) > T,\ and}$$

$$\mathbf{O_i \in ODD\ if\ S(O_i) \leq T}$$

The challenge is that there is no additional information as to how the classification is to proceed. Thus, a problem of visual oddity becomes how to determine the function S() and the threshold T.

Oddity as Analogy

Let us consider the function S(). I am addressing a problem of classifying a finite set of objects into two mutually exclusive sets. I then may reinterpret S() as being a function of two variables instead of one, and allow S() to consider a given object in terms of some or all of the other objects:

$$\mathbf{S(\ O_i, X\ )\ \ where\ \exists\ j,\ 0 < j \leq n,\ X = \cup\ O_j\ \therefore\ X \subseteq M}$$

Notice that I can further restrict X such that it contains only one object O_j:

$$\mathbf{S(\ O_i, X\ )\ \ where\ \exists\ j,\ 0 < j \leq n,\ X = \{\ O_j\ \}}$$

As I will be using S() to determine whether an object O_i is classifiable as a member of the set { O_j }, I suggest that S() represents in some sense how similar O_i is to the members of set { O_j }. In such a manner, I can proscribe that S() attain a value between 0.0 and 1.0, where a value of 0.0 would indicate entirely dissimilar, and a value of 1.0 would indicate completely similar. This characterization is important, for it allows the threshold value T to be bound to a range of from 0.0 to 1.0.

One could also infer that the value of S() being equal to 1.0 would suggest that an object was being considered for membership with a set containing itself as the set's sole element. Indeed, a value of S() equal or nearly equal to 1.0 would by this reasoning suggest that the members of the compared set would share many characteristics in common with the compared object O_i, and from that it might be inferred that the members of the set would share a similar many characteristics in common with one another. It is from these two inferences (commonality between O_i and the set, and commonality between members of the set itself) that I find the basis for comparison, that is, how the commonality is calculated.

However, note that the calculation of that similarity is yet unconstrained. Let us now consider how to restrict the calculation, and specify the role that the fractal representation can play in that restriction.

The Two Relationships

To deem some apprehended object as odd or novel involves the complex interplay of at least two relationships (Wagemans et al., 2012a): the relationship between the observer and the observed, and the relationship between the observed and its context. The relationship between the observing agent and the observed object may vary depending upon some act taken by the observer. For example, if one wishes to appreciate an object at a higher level of detail, one might move closer to the object, or bring the object closer, resulting in the object occupying a larger expanse of the observer's field of view. This action modifies the resolution of the object: at differing levels of resolution, fine or coarse details may appear, which may then be taken into the consideration of the novelty of the object. The observed object also is appreciated with regard to other objects in its environment. Comparing an object with others around it may engage making inferences about different orders of relationships. One may begin at a lower order but then proceed to higher orders if needed. The context also sanctions which aspects, qualities, or attitudes of the objects are suitable for comparison.

Given the importance of perceptual novelty detection, there has been quite a bit of work on the topic. Markou & Singh (2003a, 2003b) review statistical and neural network techniques for novelty detection. Neto & Nehmzow (2007) illustrate the use of visual novelty detection in autonomous robots. Work on spatial novelty and oddity by Lovett, Lockwood & Forbus (2008) centered on qualitative relationships in visual matrix reasoning problems. They showed that by applying traditional structure-mapping techniques (Gentner, 1983) to qualitative representations, analogical reasoning may be used to address problems of visual oddity; however, they did not show where the representations come from (Indurkhya, 1998). More recently, Prade and Richard (2011, 2013) present a logical axiomatic approach to the problem.

A General Strategy for Visual Oddity

A strategy for solving visual oddity problems should address both aspects of novelty detection described by Wagemans et al. (2012a). I now present the derivation of one such strategy.

Modeling the relationships

In my research, I model the relationship between the observer and the observed by starting with fractal representations encoded at a coarse level of resolution, and then adjusting to the right level of resolution for addressing the given problem. I model the relationship between the observed and its context by searching for similarity between simpler relationships, and then shifting its searches for similarity between higher-order relationships. In each aspect, these adjustments are made automatically by my strategy, by characterizing the ambiguity of a potential solution.

Connecting the relationships to the problem

Let us consider the second relationship (between the observed and its context). One way to judge the commonality of one object to another is to consider how its relationship between two objects compares to other such relationships. Let us suppose there is an object O_i, selected from a group of N such objects. I can denote the relationship between O_i and another object O_j in this fashion:

$$\mathbf{R(\ O_i,\ O_j\ )}$$

I model the relationship between O_i and O_j as a mutual fractal representation:

$$\mathbf{R(\ O_i,\ O_j\ ) \leftarrow T(\ O_i,\ O_j\ ) \cup T\ (\ O_j,\ O_i\ )}$$

where $T(\ O_i, O_j\ )$ is the fractal representation of O_i in terms of O_j and $T(\ O_j, O_i\ )$ is the fractal representation of O_j in terms of O_i. In this fashion, a set of relationships (indeed, they are now representations) can be determined from the entire set of objects M by considering all of the possible subsets of M which contain exactly two members.

Mathematically, if the cardinality of M is n, then the formula for the 2-combination of the set M is:

$$\mathbf{C(n,2)} = \binom{\mathbf{n}}{\mathbf{2}} = \frac{\mathbf{n!}}{\mathbf{2!(n-2)!}}$$

Thus, if there are 9 objects in the set M, then a total of 36 possible 2-combinations of M would yield 36 relationships defined as mutual fractal representations. Furthermore, it is easy to see how the number of relationships could be extended for 3-combinations, 4-combinations, or higher, through the use of the mutual fractal representation. In this manner, a strategy which uses the mutual fractal representation to model the relationship between two or more objects captures the second relationship required of noticing oddity.

The first relationship of Wagemans et al. (2012a), the relationship between the observer and the observed, can be captured as noted by varying the level of abstraction (or resolution) which at which the fractal representations are calculated. Thus, a strategy which employs fractal representations captures both relationships. However, as the strategy thus far presented considers and affords comparison between relationships, and not between objects, additional reasoning must occur.

From relationships to objects

To determine which of the objects is the most novel, the strategy must determine how dissimilar each object is from its fellows. As each of the objects has been placed into a variety of fractal representations, the model must first determine how similar each of these representations is to all of the others, and then distribute that similarity to the objects.

To calculate a measure of similarity, let us use the Tversky metric described elsewhere in this dissertation, which provides a comparison of the number of fractal features shared between each pair member (Tversky, 1977). However, in order to favor features from either image equally, and distinguished from the work in visual similarity presented earlier, here I choose to set the weights α and β equal to 1.0. Thus, I calculate similarity as a Jaccard similarity:

$$\textbf{Similarity of A and B} = \mathcal{F}(A \cap B) / \mathcal{F}(A \cup B)$$

where as before $\mathcal{F}(X)$ denotes the number of features in the set X.

Relationship Space

As this calculation for each relationship is performed, the model determines a set of similarity values for each member of this collection of fractal representations. I consider the similarity of each analogical relationship as a value upon an axis in a large "relationship space" whose dimensionality is determined by the size of the initial set. For example, if the set M had 9 objects, then for 2-combinations, the space is 36 dimensional. For relationships of 3-combinations, the space would be 84 dimensional; for relationships of 4-combinations, the space is 126 dimensional.

Maximal Similarity as Distance

To arrive at a scalar similarity score for each object of M, the model constructs a vector in this multidimensional relationship space and determine its length, using a Euclidean distance formula. The longer the vector, the more similar two members are; the shorter the vector, the more dissimilar two members are. As the model is looking for the object which is most novel, then it should seek to find the shortest vector, as an indicator of dissimilarity.

Distribution of Similarity

From the similarity score for a relationship between objects given as a mutual fractal representation, the model determines individual object scoring, the $S(O_i)$ value, by distributing the similarity value equally among all objects participating in the relationship. If an object is one of the two objects in a 2-combination, as an example, then the object's similarity score receives one half of the 2-combination's calculated similarity score. Once all similarity scores of the relationships have been distributed to the objects, the similarity score for each object is known. Algorithm 7.1 provides an overview of this similarity distribution.

Given a set of objects M = { O_1, ... O_n } and a set of representations R = { R_1, ... R_l }, where each R_i is the mutual fractal representation between two (or more) of the objects selected from M.

Let Q be a vector of cardinality |M|, initialized to 0:
$Q \leftarrow 0, |Q| = |M|$

For each representation $R_i \in R$:

- **Let S be an vector of cardinality |R|, initialized to 0:**
  $S \leftarrow 0, |S| = |R|$
- **For each representation $R_k \in R$:**
  - If i = k, then $S_k = 1 \because R_i$ is identical to itself
  - If $i \neq k$, then calculate S_k using the Tversky/Jaccard formula:
    $S_k \leftarrow \mathcal{F}(R_i \cap R_k) / \mathcal{F}(R_i \cap R_k)$
- **Let V be a scalar value, set to the normalized magnitude of S:**
  $V \leftarrow \|S\| / |S|$
- **For each object O_j which is represented by R_i:**
  - $Q_j \leftarrow Q_j + V$

The vector Q then contains the distributed similarity of each object to one another.

Algorithm 7.1. Similarity distribution

Thus, I have a general strategy for determining which object is novel. The strategy begins with a determination of the constituent objects. Each of these objects must be somehow represented in a fashion which affords its comparison to its fellow objects, in this case, via mutual fractal representations. Next, reflection occurs over those comparisons, to see if one of the objects' comparisons might be exceptional. If no one object thus stands out, perhaps reexamining some or all of the objects might be in order. Upon the conclusion of this iteration of reflection and reexamination, the object which is the most novel may be indicated.

Visual Oddity, Fractally

My model for tackling visual oddity problems consists of three phases: segmentation, representation, and reasoning. I shall illustrate the technique by working through the example problem shown above.

<u>Segmentation Phase</u>

First, the model must segment the problem into its constituent objects, which shall be labelled O_1 through O_9. In this example, the problem is given as a 478x405 pixel JPEG image, in the RGB color space. The objects are arrayed in a 3x3 grid within the problem image. At this resolution, I have found that each object fits well within a 96x96 pixel image, as may be seen in Figure 7.2.

Note that even though these objects appear to contain regular geometric shapes, the algorithm does not interpret them as such, and due to the nature of the JPEG compression algorithm, each object contains a certain quantity of noise and image artifacts. There is no processing of these images in any fashion to remove these artifacts: the pixels are addressed as received.

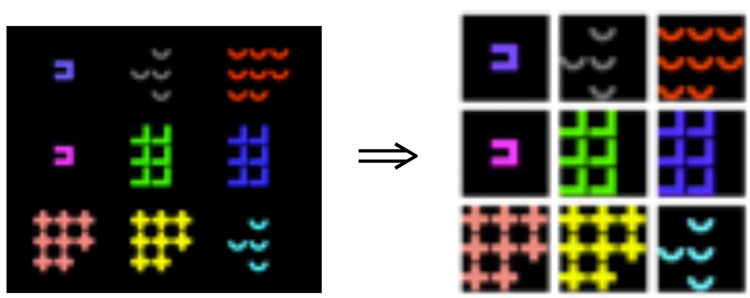


Figure 7.2. Segmentation of a visual oddity problem

<u>Representation Phase</u>

Given these nine objects, the strategy now groups objects as 2-combinations, into pairs, such that each object is paired once with the other eight objects, to form the 36 distinct 2-combinations. The strategy then calculates the mutual fractal representation R_{ij} for each pair of objects O_i and O_j, as described above.

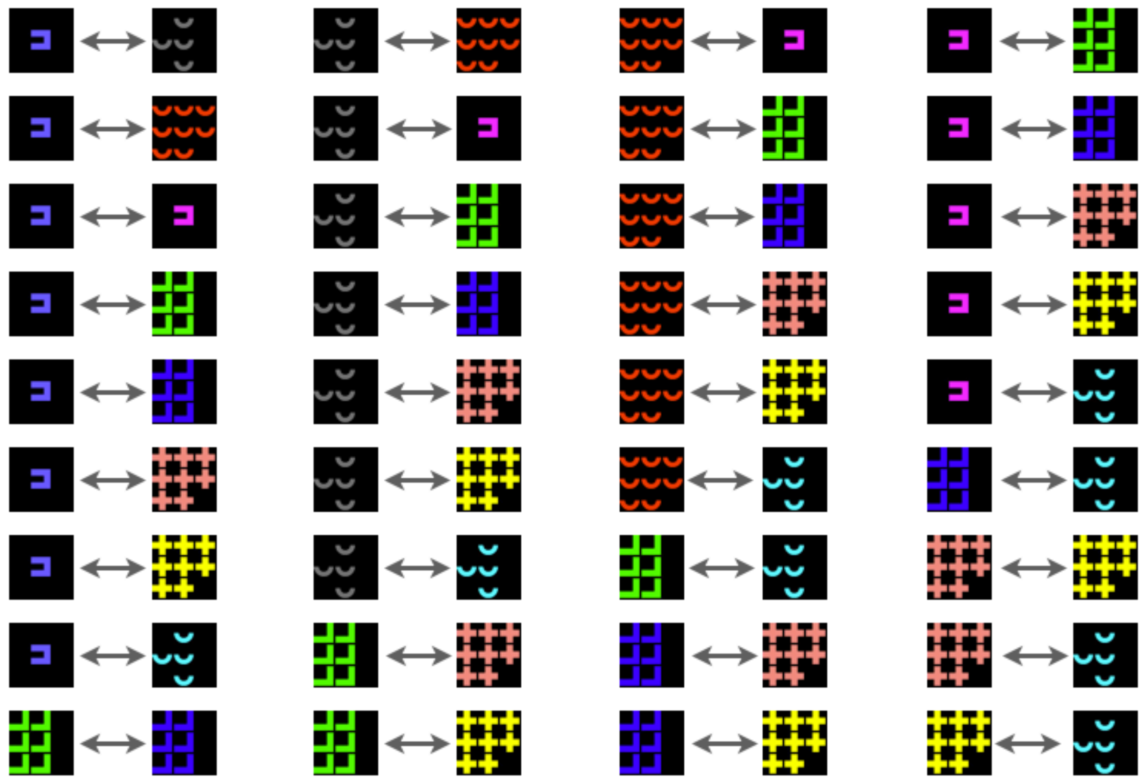


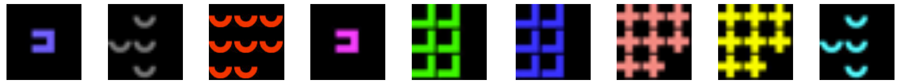
Figure 7.3. The 36 pair-wise relationships to be represented fractally

The block partitioning used initially is identical to the largest possible block size (in this case, 96x96 pixels). For this example, the strategy shall conduct the finer partitioning by uniform subdivision of the images into block sizes of 48x48, 24x24, 12x12, 6x6, and 3x3 pixels.

<u>Reasoning Phase</u>

To determine the novel object, the model must determine how dissimilar each object is from its fellows. This is accomplished, using the distributed similarity calculation described above. Accordingly, these are the distributed similarity values derived for the objects:

Table 7.1. Example similarity distribution for the initial 96x96 partition



21.301	20.206	20.384	21.301	21.985	21.814	20.639	20.632	20.198

<u>Concluding the example</u>

Once that distribution is accomplished, the model must examine the resulting similarity values to see if any object has a substantially lower score than the others. A tempting way to accomplish this would be to simply select the object with the lowest similarity distribution, but as this example illustrates, the distinction between the lowest and next-lowest score may be quite fine, and it is difficult to determine whether that distinction is substantial enough to warrant the decision.

The way the model makes this assessment is the same manner in which the model for visual similarity makes its determination: to calculate the mean of the similarity values and the standard deviation of each of the scores from this mean, and then check whether any of these standard deviations falls below the mean sufficiently to indicate a desired confidence interval (assuming a normal distribution). For this example, at a 96x96 partition, the mean score is 20.94 and the standard deviation is 0.68. If one desires a confidence of 90%, then some score must be less than 20.15 to be sufficiently novel. At this partitioning, none of the objects manages to achieve this level. If the confidence is relaxed to 80%, then the score must be less than 20.24, a value which two of the objects are calculated to be below.

Ambiguity, Abstraction, and Refinement

As the example illustrates, it is not quite so simple to select the novel object, as ambiguity may be present. Let us now be much more precise.

Ambiguity

Similarity scores for objects may vary widely. If the score for any object is "unambiguously smaller" than that of any other object, then one may deem that object to be novel. By unambiguous, I mean that there is no more than one score which is less than some ε, which may be varied as a tuning mechanism for the algorithm. I see this as a useful yet coarse approximation of the boundary between the similar and the dissimilar in feature space, the T value noted above. As I have shown, one way to characterize ε is as a confidence interval, and indeed this value may be chosen arbitrarily. I believe it to be rigorous to report the novelty as "object X, with Y% confidence." In my implementation, I use 90% as the de -facto confidence interval.

How might these ambiguities be characterized? If no object's similarity value is sufficiently lower than the mean to fall below the confidence threshold, then perhaps the value itself is derived from a set of data which is too homogenous or too sparse. If more than one object's similarity value meets the criteria, then it may also be said that the data used was too sparse or too consistent.

Abstraction

I argue then that the ambiguity arises due to a data problem, but it is more: it is a problem with the representation itself, from whence the data arise. If the data is sparse, more of it can be created; if it is too homogenous, the strategy can change how data is created, potentially affording variance.

Since the model is performing reasoning afforded by the fractal representation of the relationship between objects, it is limited in mechanisms to those which the representation sanctions. There are two primary sanctions of the representation: the number of fractal codes which constitute the representation, and the creation of features from those fractal codes.

The number of fractal codes in a particular fractal representation is determined solely by the partitioning scheme chosen when constructing the representation. The twin key observations of images which entailed fractal encoding (repetition and similarity at different scales) may be exploited here. In essence, partitioning is a modeling of how coarsely or finely an image is received or regarded, and that granularity determines the algorithm's ability to capture within the representation any present repetition or inherent similarity at that limit. The partitioning affords a level of visual abstraction. Figure 7.4 illustrates how changes in partition may be thus interpreted.

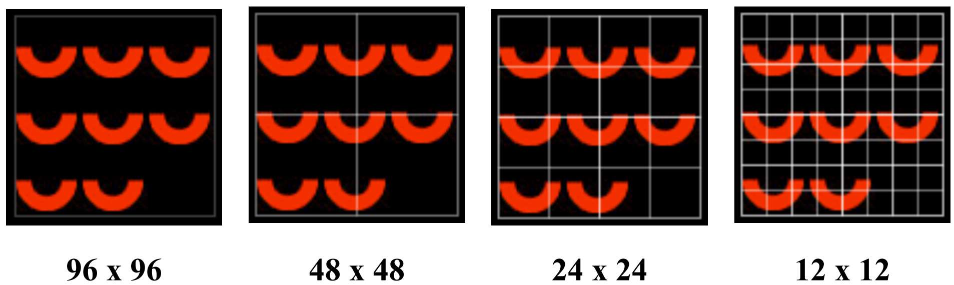


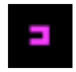
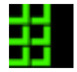
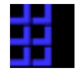
Figure 7.4. Visual abstraction as partitioning

Increasing the partitioning accomplishes two acts: more fractal codes are created, and the possible variety of features arising from those codes increases. Both of these may address the ambiguity illustrated in the data.

<u>Refinement</u>

Let me now revisit the example, and illustrate the effect of partitioning as abstraction. I redetermine the fractal representation at partitioning levels of 96x96, 48x48, 24x24, 12x12, 6x6, and 3x3 pixels successively. After each partitioning, Algorithm 7.1 is run, and the similarity distribution and the attendant confidence scores determined. Table 7.2 illustrates the result of these iterations.

Table 7.2. Confidence scores for various partitions

									
96x96	0.404	-0.719	-0.586	0.404	0.876	0.801	-0.342	-0.349	-0.725
48x48	0.840	0.304	-0.736	0.820	-0.479	-0.235	-0.744	-0.510	0.546
24x24	0.731	0.307	**-0.911**	0.853	-0.004	0.202	-0.646	-0.579	0.182
12x12	0.780	0.506	-0.657	0.810	-0.040	0.026	-0.777	-0.823	0.246
6x6	0.770	0.635	-0.269	0.722	-0.439	0.033	-0.814	-0.855	0.366
3x3	0.742	0.645	-0.016	0.687	-0.463	-0.142	-0.846	-0.847	0.458

There is one value, at a partitioning of 24x24, which yields a single answer whose confidence value exceeds 90%. The novel object for this example is therefore object 3.

A closer examination of this table reveals further nuances in abstraction. As the partitioning becomes finer, there is a moment at which the ambiguity is resolved. However, as the partitioning surpasses that point, the answer becomes ambiguous once again. In this example, other objects arise as potential candidates (note objects 7 and 8 at 6x6 and 3x3 in particular), but none exceed the confidence threshold. As the resolution reaches its limit (for these purposes, the 3x3 partitioning), there are two candidate answers, and thus ambiguity remains, even though the confidence for either candidate approaches 85%.

Ambiguity

As the level of abstraction becomes finer, the number of fractal codes, and thereby the number of features, rises. It is reasonable to presume that not all of these features will be unique at any particular level. At 96x96, there is but a single fractal code per object, and 106 features. As noted earlier, the number of features are determined by a choice of which of the primary aspects of the fractal code to use. In this example, the number 106 is arrived at by considering most (but not all) of the possible combinations of six features, bound into string structures. At 24x24, there are 16 fractal codes (1,696 features). At the finest level, there are 1,024 fractal codes (108,544 features). Why is it that ambiguity appears to resolve at a certain level of granularity, only to retreat at others?

As resolution increases, the fractal codes which represent areas in the object are covering ever smaller areas. These areas become increasingly more homogenous, and therefore the fractal codes become more similar to one another (that is, their features become more consistent). As the abstraction grows finer, more codes are devoted to representing areas of consistent color and texture. Even though the number of codes and features is increasing, the ability to discriminate based on features is decreasing. I believe this equates to a frequency apprehension of the image, with coarse resolution corresponding to low frequencies (fundamentals), and fine resolution corresponding to high frequencies (overtones, and then noise).

Thus, the disappearance and reemergence of ambiguity is an emergent characteristic of the fractal representation itself. The strategy to determine novelty is determined solely by data arising from reasoning sanctioned by the representation. In doing so, this strategy expresses the relationship between the observer and the observed.

Summary

I have shown, through this example, that ambiguity may be resolved through repartitioning, and that a strategy may be derived which notices the need for repartitioning in an automatic fashion. There exists a case which bears brief further discussion: what if every level of detail or repartitioning results in continued ambiguity?

I believe that a second strategy is to use not just pairs of objects in the calculation of similarity, but to extend that grouping to triplets or even quadruplets of objects. Through the use of extended mutual fractal representations as proscribed above, the algorithm for similarity distribution readily extends to accommodate any degree of groupings of objects without loss of generality or modification.

It is this second strategy, of extended mutuality, which captures the relationship between the observed and its context. Like the first strategy, this strategy also follows directly as an emergent consequence, an affordance, of the fractal representation.

CHAPTER 8

FRACTALS AND THE ODD ONE OUT

This chapter discusses the Odd One Out problem set, and illustrates how the visual oddity computational model works against the almost 3,000 problems in the corpus.

Odd One Out Problems

General one-one-out (or odd-man-out) tasks can be presented with many kinds of stimuli, from words, colors, and images, to sets of objects. Minimal versions of these tasks are presented with three items, from which the "odd" one must be selected. Three item one-one-out tasks, in contrast to two-item response tasks, evaluate a participant's ability to compare relationships among stimuli, as opposed to just comparing stimuli features. It has been shown that these relationship-comparison tasks track general IQ measure more closely than do two-item tasks, and this tracking of IQ increases with the number of relationships to be considered (Diascro and Brody, 1994).

Ruiz (2009, 2011) investigated odd-one-out tasks in an effort to understand and categorize an individual's g-factor (Spearman, 1904). Ruiz developed an oddity test (the Ruiz Absolute Scale of Complexity Management, or R-ASCM), and collected data from 186 university students. In particular, Ruiz sought to determine a ratio scale for fluid intelligence, based upon an interpretation of entropy as a correlate of the complexity of the problem (Ruiz, 2009). This entropy interpretation led Ruiz to develop a system which classified such problems based upon repetition of features (Ruiz, 2011).

The Hampshire Odd One Out Test

I have chosen the Odd One Out test developed by Adam Hampshire and colleagues at Cambridge Brain Sciences (Owen et al. 2010). This particular test consists of almost 3,000 3x3 matrix reasoning problems organized in 20 levels of difficulty, in which the task is to decide which of the nine abstract figures in the matrix does not belong (the so-called "Odd One Out"). Figure 8.1 shows a sampling of the problems, illustrating the nature of the task, and several levels of difficulty.

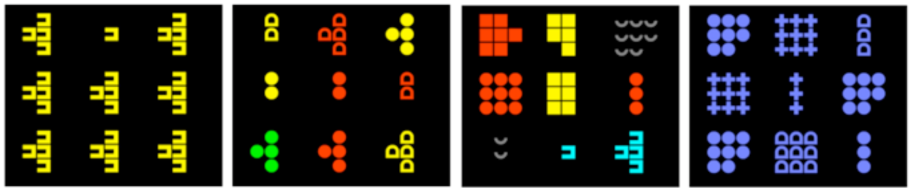


Figure 8.1. Odd One Out problems.

As is the case in most odd-one-out tasks, the matrix-like arrangement in these Odd One Out problems is arbitrary; that is, the "Odd One Out" is odd no matter the configuration. The problems are presented in that arrangement because the original data set had them so.

An algorithm for determining the Odd One Out

In the previous chapter, a general strategy for solving visual oddity problems was developed, and an example was given showing the application of the strategy to a visual oddity problem. That example problem was taken from the Odd One Out problem set.

I now present an algorithm which encapsulates the general strategy as a means for direct application to the Odd One Out problem suite. Like the strategy, the algorithm consists of three phases: a preparatory phase in which the problem image is segmented into the constituent objects, and represented as mutual fractals to afford comparison; a reasoning phase, to provide reflection over those comparisons, to see if one of the objects' comparisons might prove exceptional; and a re-representation phase, should no one object stand out,. In practice and as shown below, the re-representation phase occurs in concert with the representation phase, as a nested loop. Thus, the two phases of reasoning and re-representation occur as a unified execution phase. Upon the conclusion of this iteration of reasoning, reflection and reexamination, the Odd One Out may be indicated.

The algorithm for visual oddity is a direct extension of the Extended Analogy By Recall (ABR*) algorithm, differing only in its use of the similarity distribution algorithm as means of calculating object similarity.

To determine from a group of objects M which of the objects is the most novel or "odd."

PREPARATORY

Let $M := \{ O_1, O_2, \ldots O_n \}$ represent a group of objects.

Let $A := \{ a_1, a_2, \ldots a_l \}$ represent an ordered range of abstraction, from most coarse (at a_1) to finest (at a_l). The cardinality of A, $|A|$, and the members of A themselves are determined according to the formula given in the chapter on Analogy and Ambiguity.

Let $G := \{ g_1, g_2, \ldots g_m \}$ represent an ordered range of complexity groupings, from 2-combinations (at g1) to (n-1)-combinations (at gm). Thus $|G| = |M|-2$.

Let E be a real number which represents the number of standard deviations beyond which a value's answer may be judged as "confident." E then is the threshold value T which groups objects into the TYPICAL or ODD sets according to their similarity value.

EXECUTION

For each complexity $g \in G$:
For each abstraction $a \in A$:

- Form a set of relationships R from the objects in M according to g and a
- Derive the set of similarity values $S := \{ S_1, S_2, S_3, S_4, \ldots S_n \}$ from the set of relationships R, using the similarity distribution algorithm
- Set $\mu \leftarrow$ mean (S)
- Set $\sigma_\mu \leftarrow$ stdev (S)/$\sqrt{n}$
- Set $D \leftarrow \{ D_1, D_2, D_3, D_4, \ldots D_n \}$ where $D_i = (S_i-\mu)/\sigma_\mu$
- Set TYPICAL $\leftarrow \varnothing$
- Set ODD $\leftarrow \varnothing$
- Distribute the objects $O_i \in M$ by this rule:
  if $D_i > E$, then ODD $\leftarrow$ ODD $\cup \{ O_i \}$
  else TYPICAL $\leftarrow$ TYPICAL $\cup \{ O_i \}$
- If $|ODD| = 1$, return the object $O_i \in$ ODD
- otherwise there exists ambiguity, and further refinement must occur.

If no answer has been returned, then no answer may be given unambiguously.

Algorithm 8.1. The Visual Oddity (VO) Algorithm.

As is the case in the derivation of the Extended Analogy By Recall (ABR*) algorithm, the Visual Oddity Algorithm (VO) attempts to isolate the novel object as being the statistical outlier. The threshold value E corresponds to a deviation from the mean equivalent to some desired confidence level.

Solving the Odd One Out

The Visual Oddity (VO) algorithm was run against 2,976 problems provided by Hampshire and associates. The problems were generated by Hampshire, using a private program written in the Python programming language, and were provided to me as a set of individual images in the .PNG format. These problems span a range of difficulty in 20 levels, from the very easiest up to the most difficult, with approximately 150 problems in each level. For example, the problem in previous chapter which I used to illustrate our algorithm is a level 16 problem. No additional information as to the nature or derivation of the levels was given or sought, save that Hampshire suggested in a private correspondence that different rules were used to generate the various levels.

The VO algorithm was coded into the Java programming language, and run on a Macbook Pro computer. The process ran for several weeks, due to the large number of fractal representations which needed to be calculated. The abstraction levels used ranged from a coarse partition of 96x96 pixels down to a fine partitioning of 6x6 blocks, 5 levels of abstraction. Only 2-combination relationships were used.

Table 8.1. Results of the VO algorithm on the Odd One Out

Level	Total	Correct
1	148	147
2	148	135
3	147	119
4	149	141
5	149	88
6	149	97
7	149	100
8	149	88
9	149	114
10	149	125

Level	Total	Correct
11	149	115
12	149	123
13	149	36
14	149	38
15	149	36
16	149	34
17	149	22
18	149	28
19	149	31
20	149	30

Performance

The overall results are that the VO algorithm solved 1,647 of the 2,976 problems. Table 8.1 presents the results broken down per level. As shown by the data, the VO algorithm solves many more at the lower (easier) levels than at the higher (harder) levels.

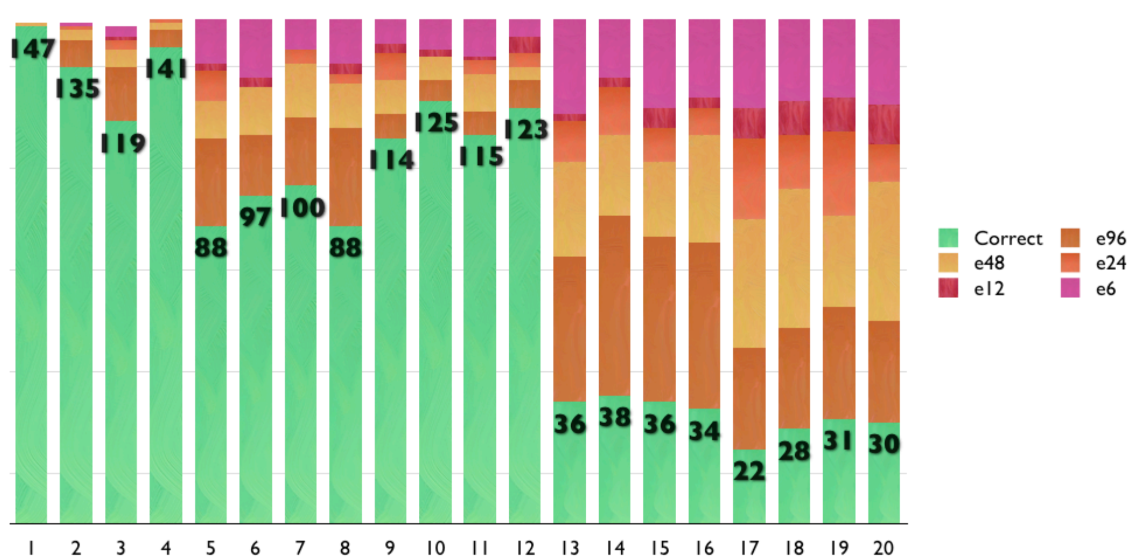


Chart 8.1. Performance and error patterns of the VO algorithm

Chart 1 illustrates the performance of the algorithm, with correct answers and subsequent error patterns noted for the various levels of abstraction. The error patterns denoted e96, e48, and so forth indicate levels of abstraction at which an incorrect answer was selected. Note that the VO algorithm stops when an unambiguous answer is reached. Thus, an e48 error pattern means that the algorithm chose an incorrect yet unambiguous answer at the abstraction level corresponding to a partitioning of 48x48.

There are quite clear degrees of performance variation generally grouped according to sets of levels (levels 1-4, 5-8, 9-12, 13-16, and 17-20). This is consistent with the knowledge that the problems at these levels were generated using varying rules. Intriguingly, it appears that the technique used by Hampshire to generate the problems at various levels enables distinct new rules at levels 5, 9, 13, and 17, which persist for the next four levels. A visual inspection of the problems did not reveal any indication of such. Therefore, the algorithm alone was able to indicate a grouping of problems. At present, the VO algorithm does not carry forward information between its execution of each problem, let alone between levels of problems. However, that the output illustrates such a strong degree of performance shift provides a further research opportunity in the areas of reflection, abstraction and meta-reasoning.

Error Patterns

As I analyzed the errors made at differing partitioning levels, I realized that most errors occur when the algorithm stops at quite high levels of abstraction. I interpret this as strong evidence that there exist levels-of-detail which are too gross to allow for certainty in reasoning. Indeed, the data upon which decisions are made at these levels are three orders of magnitude less than that which the finest partitioning affords (roughly 100 features at 96x96 versus more than 107,000 features at 6x6). I find an opportunity for a refinement of the algorithm to assess its certainty based upon a naturally emergent artifact of the representation. Although it has been shown practical to proceed with problem solving at the most coarse degree of abstraction, it may be unwise to do so.

The errors which occurred at the finest level of partitioning (the error pattern denoted e6) were caused not due to the algorithm reaching an incorrect unambiguous answer; rather, the algorithm was unable to reach a sufficiently convincing or unambiguous answer. This effect is especially noted at Level 13 and above.

These results are based upon calculations involving considering shifts in partitioning only, using 2-combinations of objects. There appear to be Odd One Out problems for which considering pairs of objects shall prove inconclusive at all available levels of detail. It is this set of problems which I believe implies that a shift in grouping (from pairs to triplets, or from triplets to quadruplets) must be undertaken to reach an unambiguous answer.

These results led me to reexamine that data in light of abstraction. From this data, I developed the theory of abstraction emergence as outlined previously. In addition, the previous analysis of errors made led me to the conclusion that too coarse a level of abstraction may lead to reasoning errors, a result I now realize (and argue above) is attributable to too sparse or too homogenous a set of features. Lastly, errors which occurred at the finest level of partitioning I now know are attributable to ambiguity, and this led me to develop and refine the use of extended mutuality in the similarity distribution algorithm, which gave rise to satisfaction of the second aspect of novelty detection as given by Wagemans et al (2012a).

Direction for future work

Future work on the Odd One Out problem should center on the interplay between the first and second strategies. I believe that it is practical, and perhaps even desirable, to explore both strategies in parallel, and allow the first unambiguous result from either strategy to be deemed the Odd One Out. Other activities to be pursued involve the use of the onset or the width of range of unambiguous partitionings as a mechanism for characterizing the ease with which a human might solve such problems: a wide range of successful partitionings might suggest an easy problem, but a narrow range, or the onset of such a range at a fine partitioning might suggest that the problem would be considered difficult.

As it is so strong an indication in the experiment conducted, a potential avenue of work is to investigate the coincidence of the failure at the extreme ends of abstraction as a signal to shift group abstraction.

CHAPTER 9

FRACTALS AND CORE GEOMETRY

In this chapter, I develop the notion that reasoning from fractal representations of visual stimuli can mimic aspects of core mathematical or geometric reasoning. I pay particular attention to a set of problems developed by Dehaene et al. (2006), and illustrate the CoreGeo algorithm, an extension of the Visual Oddity algorithm. I contrast the performance of the CoreGeo algorithm to that of another computational approach.

Mathematical Reasoning and the World Around Us

Where does mathematical ability come from? Does geometry constitute a core set of intuitions present in every human, regardless of language or education?

The first of these questions was poised by Lakoff and Núñez (2000) as a way of initiating the study of mathematics from a cognitive science perspective. Their arguments, principally that the embodied mind of humans creates mathematics and therefore it is subject to analysis in cognitive science methodologies, suggest that there may be innate principles in the mind which afford mathematical and spatial reasoning capabilities. From my representationalist point of view, I take this to mean that there are representations which the mind uses which afford these kinds of reasoning. Lakoff and Núñez discuss number discrimination in babies, and in particular look at subitizing (the ability to determine the number of objects presented from a glance), drawing on the work of others (among them, Mandler and Shebo 1982, and Trick and Pylyshyn, 1993, 1994), to note that subitizing is not a pattern recognition process. They point to Dehaene's work with patients who have suffered injury which prevents them from attending to things in their environment in a serial fashion (and therefore cannot count them), but nonetheless perform limited subitizing (Dehaene & Cohen, 1994, 1996). Obviously, statements such as these, and others made in their book have attracted much criticism to Lakoff and Núñez. For example, Schiralli and Sinclair (2003) appear to take issue with the use of metaphor rather than pattern recognition as the basis of the Lakoff and Núñez argument, citing repeated examples of the derivation of mathematical principles precisely due to the discovery of patterns. Insofar as I am aware, the debate of whether the embodied mind gives rise to mathematics, or whether mathematical principles are otherwise transcendent continues without resolution.

The second question above, on whether humans possess a core geometric intuition, is due to Dehaene (2006). Dehaene and colleagues designed and conducted a study of spontaneous geometrical knowledge of the Mundurukú, an Amazonian indigene group. The study looked at two nonverbal tests designed to probe conceptual primitives of geometry. It is the first of these tests, inspired by a test administered in an earlier study by Franco and Sperry (1977) of the hemispheric localization of geometric processing in patients with a surgical disconnection between the left and right hemispheres of their brain, that is of interest here. This particular test was designed to probe the Mundurukú's intuitive comprehension of the basic concepts of geometry, including points, lines, parallelism, and the like (Dehaene et al., 2006). For each of these concepts, Dehaene et al. designed an array of six images, five of which incorporated some desired concept, while the sixth image did not. In essence, each of these problems was a test of perceiving visual oddity.

Dehaene et al. (2006) report that the Mundurukú faired very well with core concepts of topology, Euclidean geometry, and basic geometrical figures, but they experienced more difficulty in detecting symmetries and metric properties. The Mundurukú faired poorly on two domains, each of which Dehaene et al. (2006) point out involve the mental transformation of one shape into another, followed by a second-order judgment about the nature of that transformation. Dehaene suggests that perhaps geometric transformations are more inherently difficult mathematical concepts, or that the detection of such transformations may be more difficult in static images. Dehaene also tested for comparison a group of American children and adults, and found that both the American group and the Mundurukú group shared the same difficulties on the task, although American adults performed at a higher overall level. This led Dehaene et al. to conclude that there exists some shared competence for basic geometrical concepts, regardless of culture (Dehaene et al., 2006).

Visual Oddity and Spatial Geometric Reasoning Tasks

In their 2008 paper, Lovett et al. describe a computational model for a visual oddity task, based on Dehaene's work (Lovett et al., 2008). Their rationale for choosing Dehaene appears to be two-fold: one, that the Dehaene study was designed to test which features people represent when they look at geometric figures in a visual scene, and two, that the methodology used in the Dehaene study was an oddity task methodology (e.g. look at an array of figures and choose the one which does not belong). Lovett and colleagues specifically examined the qualitative nature of the representations of the figures in the oddity task, focusing on two core claims: that when people encode a visual scene, they focus on qualitative attributes and relations of objects in the scene (therefore an abstract and robust representation) rather than a quantitative representation of the scene itself (cf. Forbus et al., 2001); and that people compare low-level visual representations using the same process as that used to perform abstract analogies. This latter claim, and the intent of their paper, is to show the versatility of the model of comparison they use, which is based on the structure-mapping theory of Gentner (Gentner, 1983).

The model described by Lovett et al. employs four systems: CogSketch to construct the qualitative representations (Forbus et al., 2008); SME to model comparison and similarity (Falkenhainer et al., 1986); MAGI to model symmetry detection (Ferguson, 1994); and SEQL to model generalization (Kuehne et al., 2000). They note that this particular model has been used in their work on the Ravens test suite, which I described in a prior chapter. The focus of their paper concerns how qualitative representations may be generated in CogSketch, but it is important to note expressly why these components were chosen and to consider the reasoning power each brings to their model.

The Structure-Mapping Engine (SME) is a computational model of the structure-mapping theory of Gentner (Gentner, 1983; Falkenhainer et al., 1986; Forbus & Oblinger, 1990). SME determines mappings between base and target symbolic representations which provide correspondences, an estimate of the similarity of the base and target, and potential inferences about the target which can be supported by the mapping and structure of the base. MAGI, based on SME, identifies symmetry by comparing a representation to itself (Ferguson, 1994), and is included by Lovett et al. (2008) to facilitate recognition of axes of symmetry in the object. SEQL (Kuehne et al., 2000) is based upon the idea that generalizations are learned through progressive alignment (Gentner & Loewenstein, 2002), that commonalities between representations are discovered as a direct result of comparison, and through a process of eliminating the portions of those representations which fail to align.

In their paper, Lovett et al. (2008) note that a key to qualitative representation is the encoding of relationships that are unlikely to have occurred by accident, ala Biederman's recognition-by-components theory (Biederman, 1987). They further distinguish qualitative aspects intrinsic to a particular shape from those discernible from the edges of the shape, but they presume that reasoning about the representations will prefer one or the other, and not both. They present twin vocabularies for describing shape attributes and relations and for describing edge attributes and edge relations. The CogSketch program is used to generate representations of the input based upon these vocabularies.

In their approach, a chosen Dehaene problem is first segmented into the individual images, and those images are then represented via CogSketch. Next, SEQL is used to create a generalization of those representations. The individual images are compared against the generalization and scored via SME. If one image is noticeably less similar to the generalization, it is deemed the one that doesn't belong. Lovett et al. note that the actual processing consists of a series of these "generalize and compare" trials, selecting subsets of the individual images from which to create generalizations. They appear to limit these subsets to be either the top three or bottom three images. Further, since their model uses representations of either the shape or the edges of the shape but not both, the system decides which version to use based upon an examination of the first image in the problem: if the image contains multiple shapes or a shape with a single edge (e.g. a circle), then the shape qualitative representation is used; otherwise, the edge qualitative representation is used. They do note that their system will abandon the edge representation if SEQL cannot generate a sufficient generalization (for example, if the images subject to generalization contain varying numbers of edges). The system looks for a sufficiently distinct candidate (they suggest a confidence of 95% as sufficient). If one is not found, then the system attempts additional trials, switching between shape and edge representation or varying the similarity scoring.

Reasoning based upon the Fractal Representation

In my research, I am exploring the extent to which the fractal representation affords analogical reasoning. The representations of the visual input given to my system represent the whole of the scene, and do not segment or otherwise seek to discriminate between objects in the scene. This represents an instant departure from the representations generated from sketches by CogSketch, in that there is no notion of either qualitative or quantitive in the fractal representation, nor is there any distinction between a shape and its edge. Furthermore, the Extended Analogy By Recall (ABR*) algorithm I developed provides a singular method of using similarity scores (chiefly determined via recall of objects indexed via features from memory) as a means both for declaring analogically derived answers as well as for providing evidence those cases in which the scene should be represented at a different level of abstraction. There are no other special purpose mechanisms (i.e. MAGI or SEQL) which augment the analogical reasoning of my system.

However, if one considers the general approach used in both my work and that of Lovett et al., the reasoning architecture itself is strongly similar. In both, the input is represented with strong commitment. In both, individual representations are compared to representations of subsets of images. In both, a sufficient score must be determined, or the models make variances in representation and try again.

For these reasons, exploring the effects of the fractal representation and the ABR* and Visual Oddity algorithms on the Dehaene problem set seemed prudent. In addition, I wished to explore the extent to which strictly visual representations and the perceptual history of an agent constructing a received world in such a manner could afford that agent rudimentary geometric reasoning capacity.

The CoreGeo algorithm

In the previous chapter on Visual Oddity, a general strategy for solving oddity problems was developed. I now present an algorithm which is derived from the general strategy expressed in the Visual Oddity algorithm as a means for direct application to the Dehaene set of core geometry problems. Like the general strategy, this algorithm, called CoreGeo, consists of three phases: a preparatory phase in which the given problem is segmented into its constituent images and represented as mutual fractals to afford similarity comparisons; a reasoning phase which reflects on those comparisons to see if any of them may prove exceptional; and a re-representation phase, shifting automatically to a different level of abstraction should no single answer stand out. In practice and as shown in the work on the Odd One Out, the re-representation phase is managed in close concert with the representation phase as a nested loop. In this way, the two phases of reasoning and re-representation occur as a unified execution phase. Upon the conclusion of the iterative reasoning, reflection and reexamination, the most visually odd item may be indicated.

The CoreGeo algorithm is a direct extension of the Extended Analogy By Recall (ABR*) algorithm and also incorporates the similarity distribution technique used above in the work on the Odd One Out problem domain as a means for calculating individual object similarity.

To determine from a group of geometrically related objects M which of the objects does not share the sought-for relationship. The specifics of the relationship are not known at the outset of the problem.

PREPARATORY

Let $M := \{ O_1, O_2, \dots O_n \}$ represent a group of objects.

Let $A := \{ a_1, a_2, \dots a_l \}$ represent an ordered range of abstraction, from most coarse (at a_1) to finest (at a_l). The cardinality of A, $|A|$, and the members of A themselves are determined according to the formula given in the chapter on Analogy and Ambiguity.

Let $G := \{ g_1, g_2, \dots g_m \}$ represent an ordered range of complexity groupings, from 2-combinations (at g1) to (n-1)-combinations (at gm). Thus $|G| = |M|-2$.

Let E be a real number which represents the number of standard deviations beyond which a value's answer may be judged as "confident." E then is the threshold value T which groups objects into the TYPICAL or ODD sets according to their similarity value.

EXECUTION

For each complexity $g \in G$:
  For each abstraction $a \in A$:
- Form a set of relationships R from the objects in M according to g and a
- Derive the set of similarity values $S := \{ S_1, S_2, S_3, S_4, \dots S_n \}$ from the set of relationships R, using the similarity distribution algorithm
- Set $\mu \leftarrow \text{mean}(S)$
- Set $\sigma_\mu \leftarrow \text{stdev}(S)/\sqrt{n}$
- Set $D \leftarrow \{ D_1, D_2, D_3, D_4, \dots D_n \}$ where $D_i = (S_i-\mu)/\sigma_\mu$
- Set TYPICAL $\leftarrow \varnothing$
- Set ODD $\leftarrow \varnothing$
- Distribute the objects $O_i \in M$ by this rule:
  if $D_i > E$, then ODD $\leftarrow$ ODD $\cup \{ O_i \}$
  else TYPICAL $\leftarrow$ TYPICAL $\cup \{ O_i \}$
- If $|ODD| = 1$, return the object $O_i \in ODD$
- otherwise there exists ambiguity, and further refinement must occur.

If no answer has been returned, then no answer may be given

Algorithm 9.1. The CoreGeo Algorithm.

The CoreGeo algorithm attempts to isolate the novel object as being the statistical outlier. The threshold value E corresponds to a deviation from the mean equivalent to some desired confidence level.

An example

Let us illustrate the workings of the CoreGeo algorithm by working through one of the problems of the Dehaene set in detail. The chosen problem is Dehaene #35, which seeks to determine an understanding of symmetry transformation about a mixed axis. The problem can be seen below in figure 9.1.

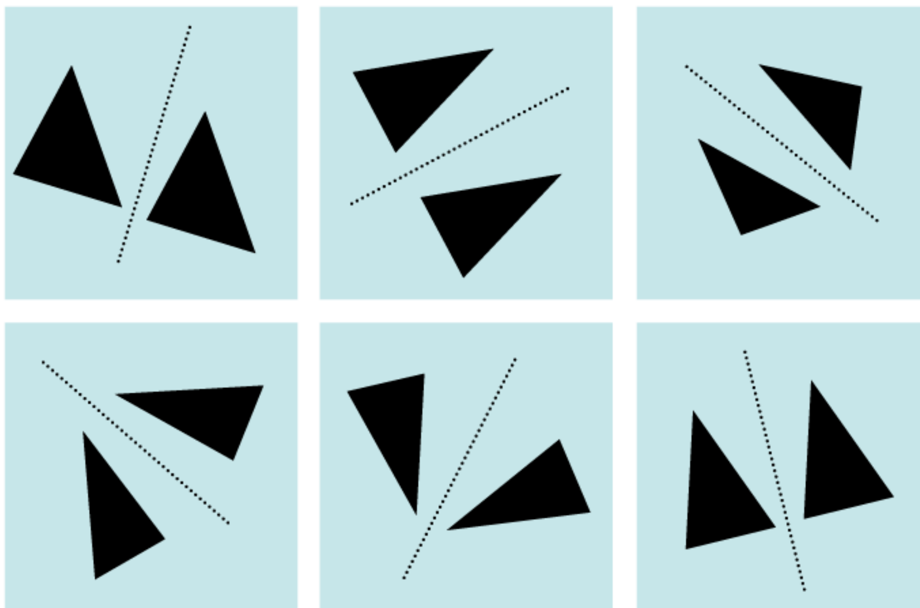


Figure 9.1. Dehaene #35, Transformation with mixed axial, symmetry.

<u>Segmentation phase</u>

First the algorithm must segment the problem into its constituent objects, which I shall label O_1 through O_6. In this example, the problem is given as a 720 x 540 .PNG image in the RGB color space. The objects are arrayed in a 3x2 grid within the problem image. At this resolution, I had found that each object fits within a 210x210 pixel image. Note that the matrix arrangement of the objects is immaterial to the problem at hand: the one which does not belong would be determined as not belonging without regard to its specific position.

As with all of the examples presented in this dissertation, even though the objects appear to be regular geometric shapes, the algorithm and the representation do not interpret them in any manner other than as mutual fractals. In addition, some noise and image artifacts are inevitably present, even though they may not be evident in the illustration here. Other than a straightforward conversion of the color image into a grayscale image (using the formula mentioned in the chapter on fractal representations), no other processing of the image to remove artifacts occurs: the pixels are addressed as received.

<u>Representation Phase</u>

Given these six objects, the strategy now groups objects as 2-combinations, pairs, such that each object is paired once with the other five objects, to form 10 distinct 2-combinations. The strategy then calculates the mutual fractal representation R_{ij} for each pair of objects O_i and O_j as described above.

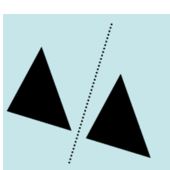
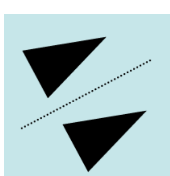
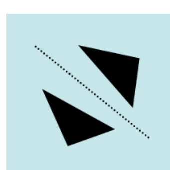
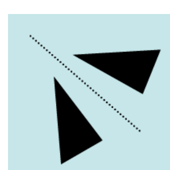
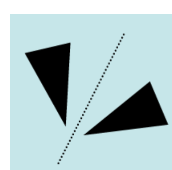
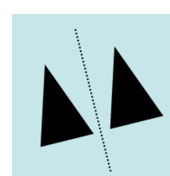
The level of abstraction used initially is identical to the largest possible pixel dimension, in this case 210x210 pixels. For this example, the algorithm shall determine the finer levels of abstraction by uniform subdivision of the images into block sizes of 105x105, 52x52, 26x26, 13x13, and 6x6 pixels.

<u>Reasoning Phase</u>

To determine the object which does not possess the same geometric relationship as the others, the algorithm must determine the dissimilarity of each object. The CoreGeo algorithm calculates the dissimilarity via the distributed similarity technique described above and in the Visual Oddity algorithm.

At the coarsest level of abstraction, these are the distributed similarity values for the objects in the example problem.

Table 9.1. Similarity distribution for the initial 210x210 level of abstraction.

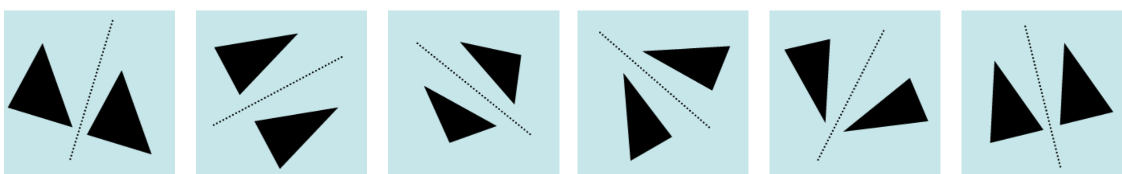
					
0.2273	0.2416	0.2303	0.2484	0.2419	0.2381

Concluding the example

Once the distribution of similarity is accomplished, the algorithm must examine the resulting values to see if any object has a substantially lower score than the others. This is wholly in keeping with the method used in the prior chapters for determining the Odd One Out.

As in the example with the Visual Oddity algorithm, the distinction between the lowest and the next-lowest score can be quite close (in this case, 0.2273 vs. 0.2303, a delta of only 0.003). If I calculate the mean of the similarity values and standard deviation of each of these values from that mean, a different picture emerges. In this example, the similarity mean is 0.2379 and the standard deviation is 0.0032. Therefore, I get the following table of deviations and subsequent confidences:

Table 9.2. Deviations and confidences for the initial abstraction level.



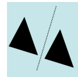
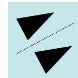
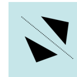
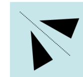
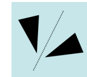
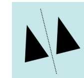
3.302	1.151	2.379	3.257	1.227	0.045
-99.9%	75.0%	-98.3%	99.9%	78.0%	3.6%

As can be seen, both the first and third answers are values well below the standard deviation and therefore at a strong confidence level. The negative values here are to indicate the least similar outliers; the strong positive confidence in the fourth answer, in contrast, indicates a strong prototype of the group. Thus, one is left with an ambiguous answer at this level of abstraction, and the algorithm must re-represent each of the 2-combinations at a finer level and try again.

Refinement

By re-representing the 2-combinations at increasing levels of abstraction, it is possible to determine an unambiguous answer. The following table illustrates the results of that successive refinement of abstraction.

Table 9.3. Mean, Standard Deviation, and Confidence for various levels of abstraction

	μ	σ_μ						
210x210	0.2379	0.003	-99.9%	75.0%	-98.3%	99.99%	78.0%	3.6%
105x105	0.189	5.5×10^{-5}	41.1%	96.3%	-39.6%	-27.3%	98.9%	-99.9%
52x52	0.271	0.005	-95.2%	-74.3%	99.9%	-65.4%	-93.9%	92.6%
26x26	0.337	0.002	-98.9%	-99.7%	99.9%	42.6%	3.7%	82.6%
13x13	0.399	0.001	-99.9%	-91.9%	99.9%	90.1%	69.6%	-60.2%
6x6	0.494	0.001	-98.3%	-99.8%	99.9%	80.0%	53.4%	2.8%

As can be seen in the table, there are three levels of abstraction for which a singular answer stands out as significantly odd, while for the other levels there exists ambiguity. If the algorithm strictly followed the philosophy of proceeding from coarsest to finest abstraction until a value stands out, then the CoreGo algorithm would select answer 6 at the abstraction denoted by 105x105 partitioning. This answer, unfortunately, is incorrect: the proper answer is answer 1. Inspection of the results shows that for all levels of abstraction except for the 105x105 level, answer 1 is among the chosen values which exceed a 95% confidence. Why would this not be true at the 105x105 level?

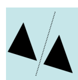
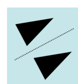
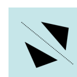
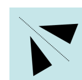
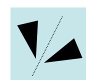
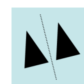
The advent of significance

A closer inspection unveils the mystery. At the 105x105 level, the standard deviation from the mean for all values is remarkably low (in this case, 5.5×10^{-5}). That deviation is two orders of magnitude smaller than all other abstraction levels. Thus, while the signal at that level of abstraction is unambiguously in favor of answer 6, the signal itself is too weak to merit consideration. In contrast, the unambiguous signal for answer 1 at the 52x52 level of abstraction is three orders of magnitude stronger.

The similarity calculations arise from the comparison of features present in the fractal representations of the relationships being examined. As noted earlier, in the chapter on visual similarity, the number of features available gives rise to the presence or absence of ambiguity, either through a sparsity of features or an increase in the homogeneity of features. In this example, I find evidence of both of these, yet the data itself has yielded now a clue for detecting homogeneity: the coefficient of variation (CV), a normalized measure of the dispersion of the values. Let us revisit the results in light of the CV for each level, calculated using this formula:

$$\mathbf{CV = \sigma_\mu / \mu}$$

Table 9.4. CV and Confidence for various levels of abstraction

	CV						
210x210	0.0126	-99.9%	75.0%	-98.3%	99.99%	78.0%	3.6%
105x105	0.0003	41.1%	96.3%	-39.6%	-27.3%	98.9%	-99.9%
52x52	0.0185	-95.2%	-74.3%	99.9%	-65.4%	-93.9%	92.6%
26x26	0.0059	-98.9%	-99.7%	99.9%	42.6%	3.7%	82.6%
13x13	0.0025	-99.9%	-91.9%	99.9%	90.1%	69.6%	-60.2%
6x6	0.0020	-98.3%	-99.8%	99.9%	80.0%	53.4%	2.8%

I now can offer an amendment to the CoreGeo algorithm, and by extension to the general models for addressing visual similarity and visual oddity. The selection of a threshold confidence value itself is not enough: the signal must be unambiguous and strong before the algorithm declares a solution. With this addition, I capture a powerful notion of analogy making and rule discovery, that the analogy must be significant enough to warrant notice.

Results of CoreGeo on the Dehaene set

The CoreGeo algorithm was run against the 45 problems provided to me by Dehaene via personal correspondence, and are the same problem set examined in Lovett et al. (2008). The problems are grouped into various categories of mathematical or geometric reasoning.

Preparation of the material

The Dehaene problem set was given as individual slides contained within a PowerPoint document. Each slide was exported into a single image in the .PNG format. Each problem image was 720 x 540 pixels, in the RGB color space. Each problem consists of six subimages, each of which upon inspection was found to fit well within a 210 x 210 boundary.

Levels of abstraction considered and calculations performed

The levels of abstraction used ranged from a coarse partition of 210x210 pixels, down to a fine partitioning of 6x6 pixels, giving 6 levels of abstraction, using the formula described in an earlier chapter for determining the maximum grid size and using a strategy of halving the pixel dimension at each successively finer level of abstraction. As the CoreGeo algorithm is a derivative of the Visual Oddity algorithm, the algorithm is capable of examining all combinations of the six subimages; however, only 2-combination relationships were examined in this experiment. This restriction was made only to serve the interests of experimental time, and illustrate the use of the algorithm on the problem set. It is important to note that the goal of the experiment was not to improve upon any prior computational model results: the goal was to illustrate that a parsimonious account could accomplish much on the test.

At each level of abstraction for each problem, the algorithm determined the similarity value as distributed amongst the six candidate images. For these calculations, the algorithm used the Tversky formula and set alpha and beta to 1.0, thus conforming the model of Gregson and Sjöberg (Gregson, 1976; Sjöberg, 1971), as it is unclear which of the difference relationships to favor. As proscribed by the CoreGeo algorithm, calculations continued until the confidence in an answer exceeded a given threshold, or until all levels of abstraction were calculated. For this experiment, that threshold value was able to be varied.

The CoreGeo algorithm was coded into the Java programming language, and run on a Macbook Pro computer. The code and entire Dehaene set of problems are available on our lab's research site, to facilitate replication of these results and future studies.

Performance

The overall results are that the CoreGeo algorithm detected the correct answer at a 95% or higher level of confidence on 35 of the 45 problems. Of the 35 problems where the correct answer was detected, 13 were ambiguously so.

The performance of the algorithm can be analyzed by varying the level of confidence required. As the table below shows, the performance of CoreGeo increases as the level of confidence required is lowered, but the ambiguity of the answers correspondingly increases.

Table 9.5. Performance of the CoreGeo Algorithm at varying levels of confidence

			confidence level					
	problem / category	**99%**	**95%**	**90%**	**80%**	**70%**	**60%**
	Total noted correct	31	35	36	38	40	42
	Correct but ambiguous	6	13	18	22	26	27
1	Training Color	yes	yes	yes	yes	yes	yes
2	Training Orientation	yes	yes	yes	yes	yes	yes
3	Topology Holes	yes	yes	yes	yes	yes	yes

4	Topology Inside/Outside						ambiguo
5	Topology Closure		ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
6	Topology Connexity	yes	yes	yes	yes	yes	yes
7	Topology Belongs To	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
8	Geometry curved lines	yes	yes	ambiguo	ambiguo	ambiguo	ambiguo
9	Geometry Convexity	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
10	Geometry straight lines	yes	yes	yes	ambiguo	ambiguo	ambiguo
11	Geometry aligned	yes	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
12	Geometry quadrilateral	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
13	Geometry right angle triangle						ambiguo
14	Geometry right angle cross					ambiguo	ambiguo
15	Geometry right angle abut	yes	yes	yes	yes	yes	yes
16	Geometry distance					ambiguo	ambiguo
17	Geometry circles	yes	yes	yes	yes	yes	ambiguo
18	Geometry center of circle	yes	yes	yes	yes	yes	yes
19	Geometry midpoint				ambiguo	ambiguo	ambiguo
20	Geometry Equilateral triangles	yes	yes	yes	yes	ambiguo	ambiguo
21	Geometry Proportion 1:3		ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
22	Geometry Diagonals	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
23	Geometry Square	yes	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
24	Geometry Rectangle	yes	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
25	Geometry Parallelogram		ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
26	Geometry Trapezoid						
27	Transformation vertical axial	yes	yes	ambiguo	ambiguo	ambiguo	ambiguo
28	Geometry vertical axial symmetry	yes	yes	yes	yes	ambiguo	ambiguo
29	Geometry horizontal axial	yes	yes	yes	ambiguo	ambiguo	ambiguo
30	Geometry random axial symmetry	yes	yes	yes	yes	yes	yes
31	Transformation translation	yes	yes	yes	yes	yes	yes
32	Transformation point symmetry			ambiguo	ambiguo	ambiguo	ambiguo
33	Transformation horizontal axial	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
34	Transformation rotation	yes	yes	yes	yes	yes	yes
35	Transformation mixed axial	yes	yes	ambiguo	ambiguo	ambiguo	ambiguo
36	Transformation homothety	yes	yes	yes	yes	yes	yes
37	Geometry Parallels	yes	yes	ambiguo	ambiguo	ambiguo	ambiguo
38	Geometry Chirality 1	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
39	Geometry Proportions		ambiguo	ambiguo	ambiguo	ambiguo	ambiguo
40	Geometry Parallels 2						
41	Geometry Chirality 2	yes	yes	yes	yes	yes	yes
42	Geometry Chirality 3	yes	yes	yes	yes	yes	yes
43	Series Arithmetic				ambiguo	ambiguo	ambiguo
44	Geometry Chirality 4	yes	yes	yes	yes	yes	yes
45	Series Geometric						

Discussion of the specific results

As can be plainly seen in the table above, there are certain problems and categories for which the CoreGeo algorithm successfully identifies a correct answer unambiguously, and others for which the algorithm is consistently ambiguous or wrong. There are still other problems and categories for which the results are mixed across the spectrum of confidence levels examined.

Topological reasoning

The CoreGeo algorithm performs well on the problems involving topological reasoning except for one type: inside vs outside. Figure 9.2 illustrates the Dehaene problem #4.

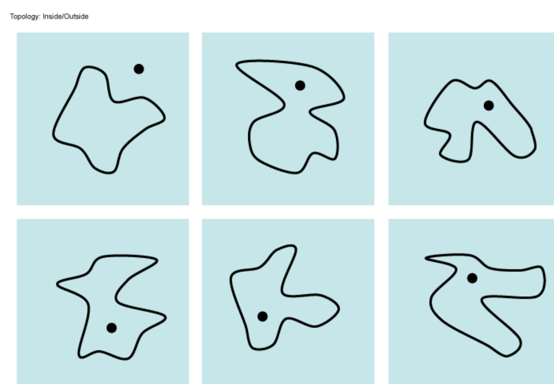


Figure 9.2. Dehaene problem #4, topological inside/outside

As in all the examples, the features over which the CoreGeo algorithm reasons are derived from the fractal representation. Between the features themselves, there is no connection, and therefore no ability to directly associate the location of the dot in the figures above as either within or without the closed line.

Geometrical reasoning

The CoreGeo algorithm performs somewhat well on problems involving geometric shapes and geometric reasoning, with two notable exceptions: reasoning about trapezoids, and reasoning about parallelism. Figure 9.3 illustrates two Dehaene problems which evoke a failing in my algorithm.

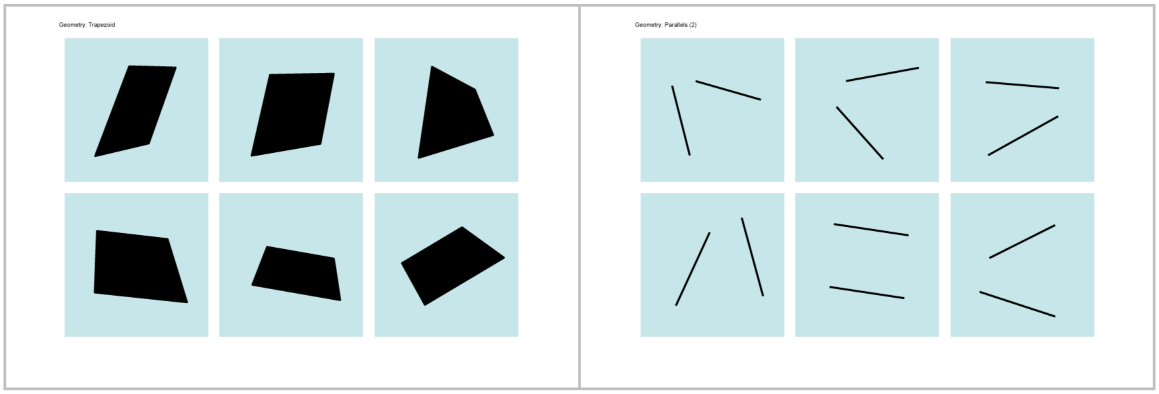


Figure 9.3. Dehaene problems #26 and #40

In the case of Dehaene problem #26, while the shape lacking non-parallel edges is apparent, at the fractal feature level, the comparisons would be with respect to finding similarity between the angles themselves. Each of the other shapes contains at least one oblique angle and two acute angles, and so the failing for this problem would seem to indicate that the numerosity of the angle kinds is absent or not readily inferred from the fractal representation. In Dehaene problem #40, as in Dehaene problem #4 above, the satisfactory answer would imply that comparisons be made between the whole line shapes, rather than their constituent parts (that is, the fractal representation of the images would note that line segments may be formed via the collage of other line segments).

<u>Reasoning about Series</u>

The CoreGeo algorithm performs poorly on problems involving reasoning about series. Figure 9.4 illustrates the two Dehaene problems which evoke a failing in the algorithm.

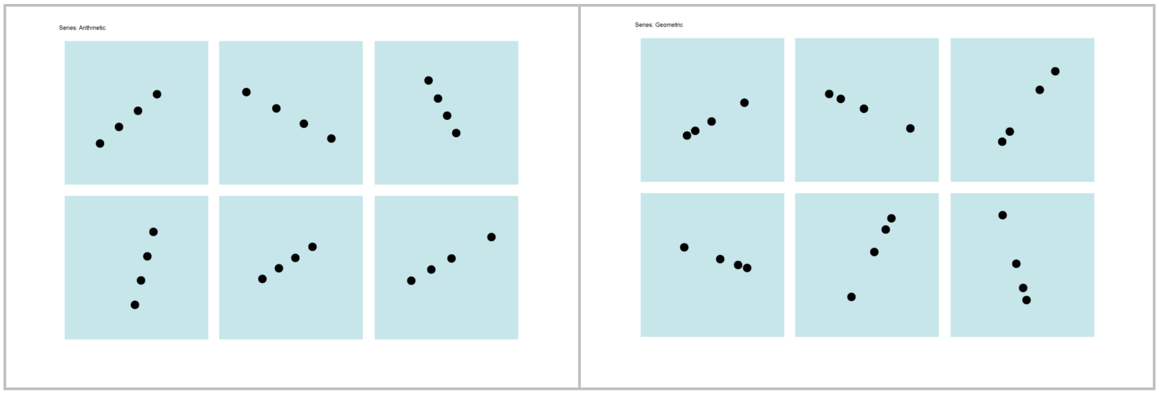


Figure 9.4. Dehaene problems #43 and #45

The CoreGeo algorithm fails to note the oddity in arithmetic progression (Dehaene problem #43) and in geometric progression (Dehaene problem #45), and in fact I should be quite surprised if it were to do so. The detection of regular progression (whether arithmetic or geometric) would require a segmentation of the image into shapes, and then a comparison of the attitude of those shapes when taken in groups of two or more. The fractal representation of a scene does not perform segmentation.

Comparison against prior efforts and human performance

Lovett et al. (2008) report that their system correctly solves 39 of the 45 problems. They also note a strong correlation between their model's performance and the performance of human test takers. Their paper presents a summarization of their results, and that of American and Mundurukú test subjects. Here is a comparison of those results with those of the CoreGeo algorithm set at a confidence level of 95%. Note that I interpret the human data as correct if the accuracy value (given in the chart in Lovett et al. 2008) is above 0.6 (about 1 standard deviation of confidence), ambiguous if between 0.6 and 0.2 (approximately random), and incorrect if below 0.2.

Table 9.6. Comparing the CoreGeo Algorithm

	problem / category	Core Geo	Lovett et al.	American	Mundurukú
	Total noted correct	35	39	43	40
	Correct but ambiguous	13	0	12	10
1	Training Color	yes	yes	yes	yes
2	Training Orientation	yes	yes	yes	yes
3	Topology Holes	yes	yes	yes	yes
4	Topology Inside/Outside		yes	yes	yes
5	Topology Closure	ambiguous	yes	yes	yes
6	Topology Connexity	yes	yes	yes	yes
7	Topology Belongs To	ambiguous	yes	yes	yes
8	Geometry curved lines	yes	yes	yes	yes
9	Geometry Convexity	ambiguous	yes	yes	yes
10	Geometry straight lines	yes	yes	yes	yes
11	Geometry aligned	ambiguous	yes	yes	yes
12	Geometry quadrilateral	ambiguous	yes	yes	yes
13	Geometry right angle		yes	ambiguous	yes
14	Geometry right angle cross		yes	yes	yes
15	Geometry right angle abut	yes	yes	yes	yes
16	Geometry distance		yes	yes	yes
17	Geometry circles	yes	yes	yes	yes
18	Geometry center of circle	yes	yes	yes	yes
19	Geometry midpoint		yes	ambiguous	ambiguous
20	Geometry Equilateral	yes	yes	yes	yes
21	Geometry Proportion 1:3	ambiguous		ambiguous	yes
22	Geometry Diagonals	ambiguous			ambiguous
23	Geometry Square	ambiguous	yes	yes	yes
24	Geometry Rectangle	ambiguous	yes	yes	yes
25	Geometry Parallelogram	ambiguous	yes	ambiguous	yes
26	Geometry Trapezoid		yes	yes	yes
27	Transformation vertical axial	yes	yes	ambiguous	
28	Geometry vertical axial	yes	yes	ambiguous	yes
29	Geometry horizontal axial	yes	yes	ambiguous	ambiguous
30	Geometry random axial	yes	yes	yes	yes
31	Transformation translation	yes	yes	yes	ambiguous
32	Transformation point		yes	yes	ambiguous
33	Transformation horizontal	ambiguous	yes	ambiguous	ambiguous
34	Transformation rotation	yes		ambiguous	
35	Transformation mixed axial	yes	yes	ambiguous	ambiguous
36	Transformation homothety	yes	yes	yes	ambiguous

37	Geometry Parallels	yes	yes	yes	yes
38	Geometry Chirality 1	ambiguous		ambiguous	
39	Geometry Proportions	ambiguous			
40	Geometry Parallels 2		yes	yes	yes
41	Geometry Chirality 2	yes	yes	yes	yes
42	Geometry Chirality 3	yes	yes	yes	yes
43	Series Arithmetic		yes	yes	ambiguous
44	Geometry Chirality 4	yes		ambiguous	
45	Series Geometric		yes	yes	ambiguous

Again, I must point out that the intention of this experiment was not to improve upon the results of the Lovett study, but to show that fractal representations and the parsimonious reasoning techniques afforded by it are capable of a fair showing. It is intriguing to note that on the six problems that the Lovett et al. model misses, the CoreGeo algorithm either answers correctly (on two of them) or ambiguously correct (on the remaining four). On the 10 problems that the CoreGeo algorithm fails to note the correct answer, one or both of the sets of human test takers score ambiguously on five of them.

CHAPTER 10

FRACTALS AND PERCEPTION

In this chapter, I present the use of the fractal representation and related algorithms as a basis for limited visual perception.

As mentioned earlier, this work on perception is presented only to show the broader utility of the fractal representation and the ABR* algorithm, and therefore contains preliminary results. The last chapter of this dissertation, on future directions, expands on methods by which this particular section may be extended or enhanced.

Agents Perceiving Fractally

While we may not be able to ascertain the workings of visual reasoning by direct interrogation, we may observe the interaction of humans and animals as they interact with each other and their environment. We might construct artificial agents, endow them with our models of such reasoning, place them into virtual worlds, and observe the correlation of their acts with their reality companions.

In nature, highly complex interactions between agents are common. Murmurations of starlings, schools of fish, and stampedes of wildebeest are at once stunning and remarkable in appearance. Even though these groups are made up of discrete individuals, the overall group splits and combines with extraordinary fluidity and grace. The collection of agents, taken together, appear to be acting under some organized control system. Yet, as Craig Reynolds, a pioneer in computer graphic flocking observes, "all evidence indicates that flock motion must be merely the aggregate result of the actions of individual animals, each acting solely on the basis of its own local perception of the world." (Reynolds, 1987) He reinforces the distinction between his work on boids and prior particle system research by remarking that to flock realistically, boids (and birds) must interact strongly with one another, and rely computationally upon both an internal state and a received external state. A simulation of flocking consists of having each agent adjust itself (modulated by internal and external state) and then rendering each agent in the simulated environment.

Reynolds' work in flocking has inspired a generation of computer graphics artists simulating natural flocking (Tu & Terzopoulos, 1994), including some of the most stunning examples ever presented on film (Allers & Minkoff, 1994; Jackson, 2003). The initial work has been extended to provide mimicry of other natural, commonplace steering mechanisms (Reynolds, 1999). In each of these systems, however, Reynolds' initial proscription for how agents interpret their environment has remained essentially intact. I now briefly introduce this proscription, as a prelude to my departure from it.

Boids, and the Three Laws of Flocking

Reynolds' boids are agents with an internal state which describes their current heading (which can be modeled by a velocity vector in two or three dimensions) and an awareness of those agents to whom they should attend (their flock mates). They also have a minimum set of intrinsic behaviors that drive them to coordinate their actions with those flock mates.

As shown in figure 10.1, the minimum set of behaviors required to produce realistic facsimiles of flocks in nature are three: stay close together, don't collide, and mimic the motion of others.

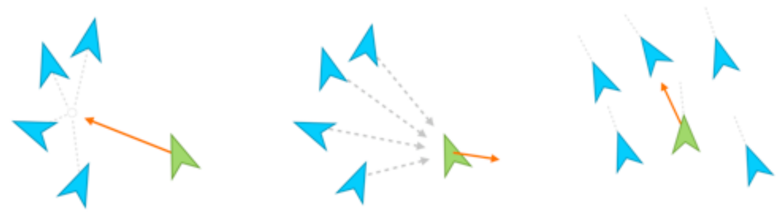


Figure 10.1. Flocking Behaviors: Cohesion, Separation, and Alignment

Cohesion

Flocking animals appear to want to be close to others like themselves. In simulations, this is achieved by calculating a centroid of the apparent position of flock mates, and adjusting a boid's heading to aim in that direction.

Separation

Animals generally do not want to collide with one another. The separation behavior balances the cohesion behavior by forcing a boid's heading away from the apparent direction of each individual flock mate.

Alignment

Animals mimic one another. A way to provide this mimicry to agents in a flock is to have each agent attempt to match the movement of its flock mates. In practice, this is accomplished by having a boid adjust its heading to align with the aggregate direction of its flock mates.

Interaction

These three behaviors interact with one another in specific ways. The separation behavior affords static collision avoidance, in that the position of flock mates is perceived at every new moment, and thus may be considered static. In contrast, the alignment behavior is dynamic, in that the heading (and not the position) of flock mates is considered. As Reynolds points out, this is a simplified predictive version of collision avoidance, complementary to the separation behavior, in that boids that mimic the motion of their flock mates are less likely to collide with them than would boids which moved freely (Reynolds, 1987). The cohesion behavior drives a boid to become the center of its flock, with the urge to move to the center modulated by its distance from the centroid of its mates. This movement to the center is localized, and allows a flock to split around obstacles (or other portions of the flock) with natural fluidity.

Perception

A flock in nature (a murmuration of starlings, for example) may be composed of many thousands of individuals. It would seem an improbable computational load to place upon each agent within the flock the attempt to ascertain aspects of each member of the flock prior to making modifications to its own behavior. Some restriction of which individuals to consider must occur. Reynolds characterizes this as considering each agent to have a local perception. In computer simulations of flocks, the local perception each agent has of the world typically is provided to the agent by a godlike view of the entire environment, and a superimposed restriction of individuals by culling those deemed too distant to consider. This distance is usually referred to as a range of influence.

The Froid World

The oraclesque decision of whom to consider is only practical in computer simulations. In natural flocks, clearly no such ability is afforded, and each animal must make its decisions based upon some combination of what it is perceiving of or thinking about its world. The choice of flock mate is crucial for the remaining behavior to succeed.

For explorations of visual reasoning, affording agents with models of perception based on familiarity and novelty and observing those agents as flocks seems ideal. Prior research has employed fractal representations to model and discover similarity and novelty in visual analogy tasks such as intelligence tests (references omitted). In my system, I wish to endow my agents with a visual reasoning apparatus that embodies precisely these characteristics. Thus, for my flocking simulation, I created agents possessed with the ability to receive their local environment by localized observation only, and to perceive this received world via manipulations of fractal representations. I call my agents froids ("fractal boids").

Froids versus Boids

The difference between my froids and typical Reynolds boids is two-fold: froids sense and then classify their environment, whereas boids are told explicitly about their surrounds. Both boids and froids manifest the same behaviors, and thus participate in flocking with their mates, but only froids perceive and reason about their environment prior to enacting those behaviors. Figure 10.2 illustrates the visual reasoning pipeline of a froid, from the reception of the world, through perceiving individuals and objects in the world, reasoning about those perceptions, and finally to enacting a decided upon course of action.

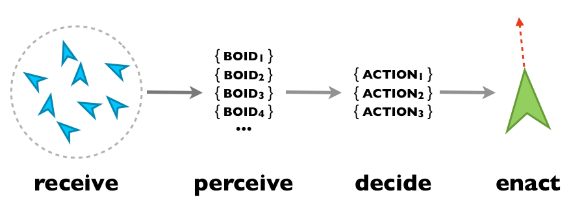


Figure 10.2. Visual reasoning pipeline

The perceptive system of a froid must be computationally efficient, to permit sufficient time to select and enact a behavior based upon the arriving stimulus. In animals, the perception system, while informed by the decision and control system, appears to operate concurrently with those systems, providing a real-time appraisal of a continuously shifting world. For the purposes of my experiment, and in the general case for systems based upon Reynolds' boids, the simulation of the agents within the world proceeds in discrete steps. Thus, the available stimuli from the world changes at a known pace, and I need not provide for parallel processing with the visual reasoning pipeline, nor for the need to interrupt an action in process to accommodate new information.

I therefore made two simplifying architectural decisions. First, the perception stage occurs in a serial fashion with the behavior decision stage, since the world of the simulation will not have changed until all the agents have moved themselves. Second, the perception stage would act only upon newly arriving stimuli, and not be influenced by prior decisions. This variety of architecture is deemed "reactive control" in robotics (Arkin, 1998) and "situated action" in cognitive science (Norman, 1993). I made these simplifications so that I might better compare the effect of perception on the subsequent behavior, without having my analysis take into account any perceptual hysteresis or other internal state.

I now shall describe each stage of the visual reasoning pipeline in some detail.

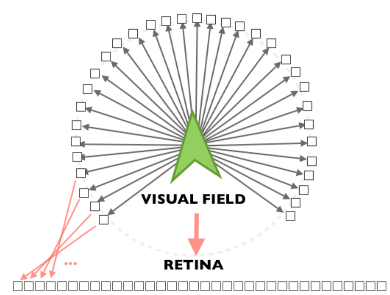


Figure 10.3. Visual field to retina mapping

How a Froid Sees

I imagine a froid as having a single "eye" with a broad field of view. The froid's eye consists of a simulated retina, an arrangement of sensors. A froid sees its environment by receiving photometric stimulation upon this retina. The light entering each of these sensors is combined to form a visual field, as shown in figure 10.3. In my simulation, I use ray-casting to send a ray out through each of the sensors into the simulated world, and note whether that ray intersects anything. I illustrate this in Figure 10.4.

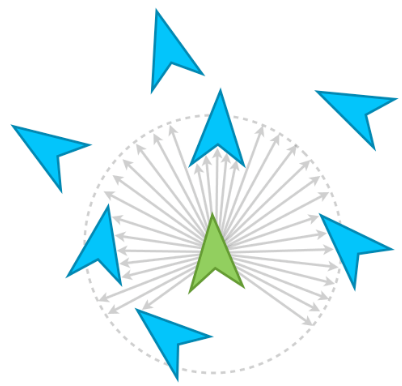


Figure 10.4. Seeing via ray casting

The froid interprets the "light" falling upon the sensor is a function of the distance of the intersected object from the froid, where objects which are distant are fainter than close objects. No characterization is made regarding what object has been intersected, only that an intersection has occurred at some distance. Figure 10.5 shows an example of how objects within the froid's immediate environment may be mapped by this visual system onto its retina.

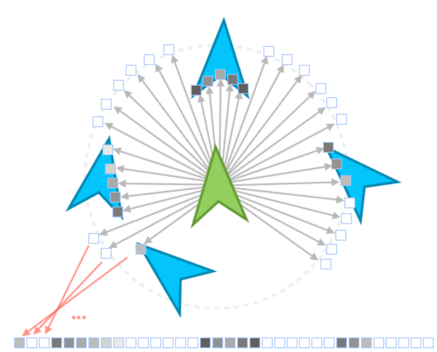


Figure 10.5. Objects in the environment, retinal image

Fractal Perception

The photometric values arriving via the froid's retina next are interpreted by the froid's perception stage. While there may be many possible objects one may wish to divine from this visual stimuli, I restrict the intentionality of the perception to only those tasks which will drive the flocking behavior. Accordingly, the primary task of the perception system is to determine flock mates.

This, however, raises an immediate question: what does a flock mate look like to a froid? My froids are rendered into the simulated environment as chevrons whose orientation, color and physical size may vary. The visual environment, as transduced onto the retinal image, will show only an arranged set of values, roughly corresponding to visual distance to whatever object happened to intersect the ray from the sensor.

Filial Imprinting

I was inspired to approach this problem using techniques from neurological and biological research. Certain baby animals acquire behaviors from their parents, via a process called filial imprinting. Implicit in the imprinting is the ability to identify a parent. There is evidence that certain species have innate or rapidly develop through acclimation visual prototypes which allow young members to accurately identify their parents (O'Reilly & Johnson, 1994).

There are many possible visual arrangements between a froid and a prototypical "other" in its environment. I chose to restrict these prototypes to six: four which corresponded to points on the compass (north, south, east and west), and two which corresponded to specific situations which would seem useful for behavior selection (close and empty). Figure 10.6 illustrates these filial imprints, along with their corresponding retinal impressions. These imprints are given as innate knowledge to each froid.

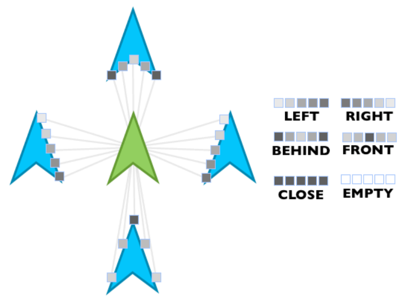


Figure 10.6. Filial Imprinting

Fractal Imprinting

A froid encodes all its visual information, its retinal data, as a fractal representation. Accordingly, each imprinted prototype is encoded into a fractal representation, and placed, indexed by derived fractal features, into the froid's memory system. The technique for transforming the image into a fractal representation and the derivation of fractal features from that representation is the same as described earlier in this dissertation.

This imprinting, encoding, and memorization provides each froid with a static knowledge base. From this foundational base, a froid can receive new retinal images and seek within those arriving images what it believes to be familiar.

Finding the familiar by visual analogy

The arriving retinal image is an otherwise undifferentiated collection of photometric information, with each value corresponding to a particular direction and distance. From this retinal image, flock mates that might be within the visual range of the froid may be identified.

As shown in figure 10.7, the algorithm begins by segmenting the retinal image into varying sets (collections of adjacent sensors), and then encoding each of these segments into fractal representations. No attempt is made to interpret the retina image for edges or other boundary conditions: the segments are treated merely as they are found.

Additionally, the segment size itself is arbitrarily chosen. For my experiment, I selected a segment size corresponding to 10 retinal sensors, with the entire retina being 90 sensors in size, encompassing a field of view roughly 135°, oriented to the froid's forward motion. Thus, each retinal image yielded nine segments for analysis.

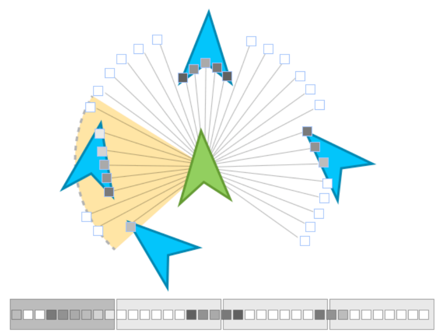


Figure 10.7. Segmenting the retinal image

The process of perceiving a segment and possibly selecting a familiar prototype is given in Algorithm 10.1. First, the segment is encoded into its fractal representation, exploiting its self-similarity. Next, the froid's memory is interrogated for similarity with imprinted prototypes, using a scoring system for featural similarity as described by Tversky (1977). The most similar imprinted prototype is chosen as the interpretation of that segment of the froid's retinal image.

To determine the prototype P’ which is most analogous to the retinal segment R from a set of fractal prototypes $P \coloneqq \{ P_1, P_2, \ldots P_n \}$:

$F \leftarrow$ Fractal(R, R)
Set $M \leftarrow 0$ and $P' \leftarrow$ unknown
For each prototype $P_i \in P$:
· Calculate the similarity of F to P_i : $S \leftarrow$ Sim(F, P_i)
· If $S > M$, then $M \leftarrow S$ and $P' \leftarrow P_i$

P’ is therefore that prototype $P_i \in P$ which corresponds to the maximal similarity S, and is deemed the most analogous to retinal segment R.

Algorithm 10.1. Selecting the familiar

If a segment appears to correspond to an imprinted prototype (and not to empty space), then several inferences may be made. The first is that an individual flock mate exists in that direction of view, which corresponds to the segment’s retinal constituents. Secondly, it may be inferred that the flock mate lies at a distance which corresponds to a function of the faintness of the photometric readings of that portion of the froid’s retinal image. By systematically examining each segment of the retina, the froid’s flock mates thus may be inferred by visual analogy.

The Three Laws for Froids

Once the flock mates have been discovered, the Reynolds rules for flocking may be invoked. Since the perception system has inferred the existence of a flock mate at a particular distance and direction, the separation and cohesion rules may be enacted directly. However, the alignment rule’s application requires further inference.

To align with a flock mate, the froid must infer the heading from the visual classification of the mate. This classification depends explicitly upon which of the filial prototypes has been selected as most representative of the retinal segment. Algorithm 10.2 provides the following five rules of heading inference.

To determine the heading H for an identified flock mate with classification C and apparent direction D:

If C = LEFT, then H ← D - 90°
If C = RIGHT, then H ← D + 90°
If C = BEHIND, then H ← D
If C = FRONT, then H ← D + 180°
If C = CLOSE or C = EMPTY, then H ← unknown

Algorithm 10.2. Inferring flock mate heading

Figure 10.8 shows an example of these inference rules at work. In this example, the retinal image is classified as most similar to the RIGHT filial prototype. The heading of this identified object is inferred to be at a 90° angle to its apparent direction. Note that the partially viewed individual does not sufficiently cover enough retinal space to be identified.

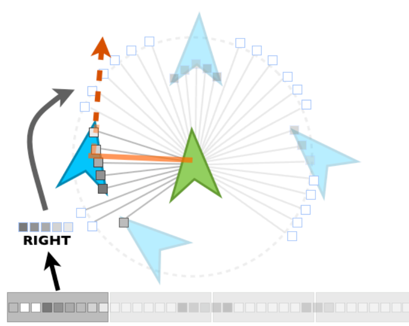


Figure 10.8. Inferring heading from a retinal segment

Once the heading is inferred, the alignment rule of Reynolds may be used to adjust the motion of the froid.

Froids and Boids

To test my belief that a froid could behave as naturally as its boid counterparts, I created a traditional Reynolds- style boid system, written in Java, running on a conventional computer system. I first placed into the environment several thousand standard boids, and observed that their aggregate motion was as expected: a realistic simulation of natural flocking behavior.

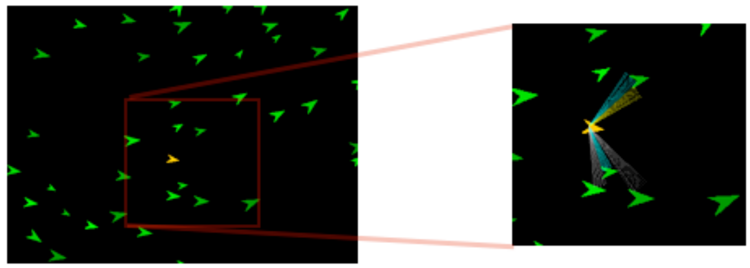


Figure 10.9. A froid flocks with boids, a closeup of the froid perceiving its environment

I then introduced one froid into the environment with the boids. Figure 10.9 shows a view of this simulation, with traditional boids in green, and the froid in gold. I observed that the froid, whose identification of flock mates was based solely upon its fractal perception system, behaved in the same manner as those boids whose identification of flock mates was given in the traditional oracle manner. I subsequently added several more froids into the mix, and found that the overall flocking behavior remained consistent and realistic. Figure 10.9 also shows a closeup of the froid, in the company of several boids, with its perception system visualized, perceiving and classifying its flock mates.

Unlike the boids, the froids appeared to suffer from uncertainty (manifested by a stuttering motion) when in the proximity of a large number of other boids. I surmised that this is due to the inability of the segmentation system using within the retina to accommodate or otherwise classify large amounts of overlapping or confounding visual data. Another possibility concerns the enaction itself. Let us suppose that two action vectors arising due to two received perceptual signals almost exactly cancel each other. In this case, small fluctuations in the perceptual signal can cause a significant change in the action vector, which may result in stuttering.

Summary

Through this experiment, I have shown that the fractal techniques previously developed for visual analogies can be used for perception. I constructed an artificial boid like world that is popular in graphics and games, and demonstrated that froids (fractal-based boids) can use the fractal technique for mapping percepts into actions in real time and manifest flocking behavior.

While the use of fractal representations is central to my technique, the emphasis upon visual recall in my solution afforded by features derived from those representations is also important. There is evidence that certain species have innate or rapidly develop through acclimation visual prototypes which allow young members to accurately identify their parents (O’Reilly & Johnson, 1994). I hold that placing imprints into memory, indexed via fractal features, affords a new and robust method of discovering image similarity, and that images, encoded and represented in terms of themselves, may be indexed and retrieved without regard to shape, geometry, or symbol. I also hold that the representations of the perceptual stimuli, as in my fractal technique, need to be built at run-time and in real-time.

CHAPTER 11

FRACTALS AND GESTALT PERCEPTION

The relationship of my computational approach to visual analogy using fractal representations and the gestalt view of visual perception in cognitive psychology (Arnheim, 1954; Steinman et al., 2000; Wagemans et al. ,2012a, 2012b) requires examination. The gestalt view of visual perception recently has received attention in computational models of visual analogy (e.g., Schwering et al., 2007, but obliquely Dastani & Indurkhya, 1997, 2000; and Ojha & Indurkhya, 2009). As in gestalt methods in general, my fractal approach to visual analogy constructs different interpretations of the input dynamically and re-represents the problem as needed.

As mentioned earlier, this exploration of gestalt perception is presented only to show the broader utility of the fractal representation and the ABR* algorithm, and therefore contains promising but very preliminary results. The last chapter of this dissertation, on future directions, expands on methods by which this particular section may be extended or enhanced.

Bistable Perception and the Necker Cube

A visual percept is deemed bistable if there are two potential yet mutually exclusive interpretations of the percept between which the human visual system cannot unambiguously choose. An additional characteristic of bistable perception is that alternation between the available interpretations appears to happen in an uncontrollable, spontaneous and stochastic manner (Kogo et al., 2011; Nagao et al., 2000). Perhaps the most famous example of a bistable visual percept is the Necker Cube, shown in Figure 11.1 (Necker, 1832).

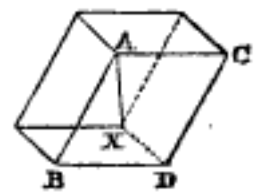


Figure 11.1. The original Necker Cube

As Necker notes concerning this illustration from crystallography, although the figure is drawn to indicate that the solid angle labelled A should be seen as closest and the solid angle X should be seen as furthest (and therefore, the face ABCD would be foremost), one's perception of the figure will shift involuntarily to cause the opposite interpretation (Necker, 1832).

Investigations of Bistable Perception

Psychological and cognitive neuroscientists have been fascinated by the advent and potential cause of bistable perception, though not all of their research has been concerned with the Necker Cube. For example, one way to induce a bistable percept in humans is to present dissimilar visual images to each eye, in a process known as binocular rivalry (Meng & Tong, 2004; Mitchell et al., 2004; Tong et al., 2006). When presented in such a manner, the images compete for perceptual dominance, with each image "available" in a perceptual sense for a few moments while the other image is perceptually suppressed. Because the changes in perception occur without changes in the actual stimuli, studies have been conducted to establish the neural correlates of those perceptual responses. Lumer et al. (1998) report that fMRI studies revealed cortical regions typically associated with spatial attention were active, but that activity in the frontopariental cortex were specifically associated with perceptual alternation, suggesting that visual awareness was biased toward abstract internal representations rather than merely the arriving percept's spatial arrangement.

That there appears to be a neural correlation between bistable perception and the regions of the brain thought to involve abstraction representation is of keen interest to my research. As ever, the question arises as to what the nature of that representation may be, and therefore what sorts of reasoning would it endorse.

Work on the Necker Cube problem has occurred since Necker's original paper (Necker, 1832) on the subject. A fair number of efforts have examined the perceptual alternation problem. Einhauser et al. (2004) performed eye tracking studies and found that there is a close link between the perception of the Necker cube and eye position, wherein a subject's eye position shifts after their perceptual shift, moving to a extreme position that then, in turn, caused a perceptual shift, suggesting that somehow eye position suppress the older percept. In examining the timing information available in EEG studies, Kornmeier and Bach (2004) found an early electrophysical correlate of the perceptual reversal in an Necker cube by comparing exogenous reversals of unambiguous stimuli to the endogenous reversal in Necker-like stimuli. This suggested to Kronmeier and Bush that the emergence of a 3D interpretation of the stimuli and its reversal likely occur in purely visual areas, but that the act of perceiving the stimuli is modulated from a higher level in the visual system, perhaps a confirmation of the earlier results of Lumer et al. (1998).

Noest et al. (2007) present a neural model of perceptual switching which exhibits the percept choosing and spontaneous switching without any high-level decision making or memory. Their work specifically addresses the problem that the ambiguous visual stimuli is viewed continually, but the perception of that constant stimuli switched many times.

Of particular consequence to my work is that the investigations of Noest et al. (2007) were guided by their recognition of the process of ambiguity resolution as an example of dynamically equivalent nonlinear processes that occur throughout nature which are mathematically well characterized (Cross & Hohenberg, 1993; Guckenheimer & Holmes, 1983). Sundareswara and Schrater (2008) likewise investigated ambiguity resolution as a potential explanation for perceptual bistability by exploring the effects of the background on the viewpoint selected.

Perceiving the Necker Cube, Fractally

As a consequence of my research, I wished to see how the Extended Analogy By Recall (ABR*) algorithm would perform when considering the Necker Cube problem. In particular, what I sought to discover was whether the algorithm would exhibit an inability to choose between alternative visual interpretations of the Necker Cube.

The input data

I set up the experiment in the following manner. First, I created a very exact rendition of the Necker Cube at a resolution of 200x200 pixels, and saved it in the .PNG format in the RGB color space. This target cube is shown in Figure 11.2.

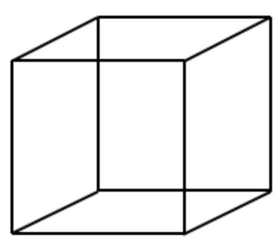


Figure 11.2. The target Necker Cube.

I then created, from that original drawing, three sets of alternative visual interpretations of the cube, each set containing two interpretation choices: C_1, an image with the forward face lowermost, and C_2, an image with the forward face uppermost.

Each set maintained the same isometric projection as the target cube, but in each set, a visual cue was embedded to suggest which face was forward. In Set 1, a technique known as “haloed lines” was used (Appel et al. 1977). In Set 2, the edges which are to be interpreted as “behind” are rendered in a slightly different color. In Set 3, the occluded edges are removed entirely, leaving an impression of a solid cube. In each set, the individual images were 200x200 pixels, and saved into the .PNG format in the RGB colorspace. Figure 11.3 illustrates the alternative pairs I created.

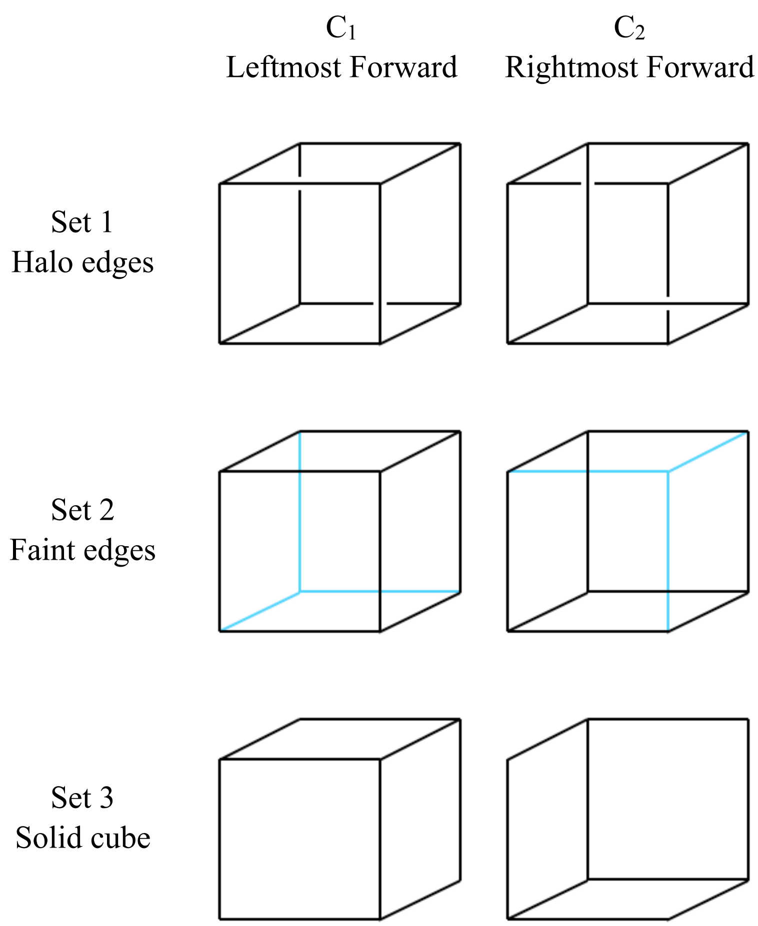


Figure 11.3. Sets of alternative interpretations of the Necker Cube

Creating the relationships

Since the ABR* algorithm compares the similarity between fractal representations, I created for each set three fractal relationships. The first relationship was between the target and itself (to establish a self-referential identity). The second relationship was between the target and the set's C_1 image, and the third was between the target and the set's C_2 image. Thus, for each set, these mutual fractals were created:

$$\mathbf{R = MutualFractal(\ target, target\ )}$$

$$\mathbf{R_1 = MutualFractal(\ target, C_1\ )}$$

$$\mathbf{R_2 = MutualFractal(\ target, C_2\ )}$$

The problem, then, becomes this: to which of the two relationships, R_1 or R_2, is the R relationship most similar? Another, and most concise, statement of the problem for each set would be: if R_1 and R_2 are known (previously experienced, kept in memory), of which is R most analogous?

Calculating Necker analogies

The algorithm for calculating Necker analogies for each set, the Fractal Necker algorithm, is a derivation of the ABR* algorithm, and is given below. As in the ABR* algorithm there are three phases: preparatory, examination and re-representation. Indeed, as in the other instances of the ABR* presented in this dissertation, the examination and re-representation phases are combined into a single execution phase for expedience.

Given a target Necker cube and set of possible interpretations, determine an answer.

PROBLEM SEGMENTATION

By examination, the set of interpretations are individual images.

Let T be the target Necker cube image.
Let C := { C_1, C_2, ... } be the set of individual interpretations.

RELATIONSHIP DESIGNATIONS

Let R be a relationship, determined as follows:
R ← MutualFractal(T, T)

ABSTRACTION LEVEL PREPARATION

Let d be the largest pixel dimension for any image in the set $M \cup C$.

Let δ be the abstraction decrement value where $1 \leq \delta \leq d$.

Let A := { a_1, a_2, ... } represent an ordered range of abstraction values where
$a_1 \leftarrow d$, and $a_i \leftarrow a_{i-1} - \delta \; \forall\, i, 2 \leq i$ and $a_i \geq 2$

The values within A constitute the grid values to be used when partitioning the problem's images.

Algorithm 11.1. The Fractal Necker Algorithm, preparatory stage.

The Fractal Necker Algorithm: Preparatory Stage

In the first stage of my Fractal Necker Algorithm, a Necker Cube problem is first segmented into its component images (the target image T, and the collection of interpretation images). Next, the algorithm determines the relationship between the target and itself, expressed as a mutual fractal representation. Then, a range of abstraction levels is determined.

As in other implementations in my research, the abstraction levels are determined to be a partitioning of the given images into gridded sections at a prescribed size and regularity. In contrast to earlier implementations of the ABR* algorithm, however, in this experiment I wished to note the circumstances under which the algorithm would prefer one or the other alternative interpretation of the target Necker Cube. Therefore, for this experiment, I allowed the level of abstraction to begin at the coarsest possible level (200x200), but decrease in a regular fashion, in steps of 3 pixels. Thus, the abstraction level proceeded as follows: 200x200, 197x197, 194x194, and so forth, down to a minimum level of 5x5.

Given M, C, R, A, and η as determined in the preparatory stage, find the answer.

PREPARATORY

Let E be a real number which represents the number of standard deviations beyond which a value's answer may be judged as "confident"

Let S(X,Y) be the Tversky similarity metric for sets X and Y

EXECUTION

For each abstraction $a \in A$:

- Re-represent each fractal representation $r \in R$ according to abstraction a
- $S \leftarrow \varnothing$
- For each answer image $c \in C$:
  $R' \leftarrow \text{MutualFractal}(T, c)$ according to abstraction a
  $S \leftarrow S \cup \{ S(R, R') \}$
- Set $n \leftarrow |S|$
- Set $\mu \leftarrow \text{mean}(S)$
- Set $\sigma_\mu \leftarrow \text{stdev}(S)/\sqrt{n}$
- Set $D \leftarrow \{ D_1, D_2, \ldots D_n \}$ where $D_i = (S_i-\mu)/\sigma_\mu$
- Generate the set $Z := \{ Z_i \ldots \}$ such that $Z_i \in D$ and $Z_i > E$
- If $|Z| = 1$, return the answer image $C_i \in C$ which corresponds to Z_i
- otherwise there exists ambiguity, and further refinement must occur.

If no answer has been returned, then no answer may be given

Algorithm 11.2. The Fractal Necker Algorithm, execution stage.

The Fractal Necker Algorithm: Execution Stage

The algorithm concludes by using a variant of the ABR* algorithm to determine the confidence in the answers at each level, stopping when ambiguity is sufficiently resolved. Thus for each level of abstraction, the relationship R is re-represented into that partitioning. Then, for each of the candidate images, a potentially analogous relationship is determined and a similarity value is calculated. The balance of the Fractal Necker algorithm follows the ABR* algorithm, using the deviation from the mean of these similarities, continues through a variety of levels of abstraction, looking for an unambiguous answer that meets a specified confidence value.

However, in this experiment, I did not allow the algorithm to halt if one or the other interpretation exceeded the confidence threshold. To achieve this, I set the confidence level artificially high (100%), a result unobtainable. This then caused the algorithm to proceed to calculate and report similarity values at all levels of abstraction.

Results of the experiments

I ran the algorithm on each of the three sets given above. The algorithm was coded in the Java programming language, and run on a Macbook Pro computer. The total running time required was less than a day, the bulk of which was taken up by the construction of the various fractal representations. As with previous algorithms and experiments in this dissertation, the code and example images are available on our research lab's website for ready replication and extension.

As indicated above, the algorithm calculated similarity values for all of the available levels of abstraction, beginning with the coarsest (200x200) and proceeding in a regular fashion down to the very finest (5x5). At each level of abstraction, the similarity value for each of the possible interpretations is calculated, using the Tversky formula, and set alpha to 1.0 and beta equal to 0.0, conforming to values used in the coincidence model by Bush and Mosteller (1953). From those values, the algorithm calculated the mean and standard deviation, and then calculated the deviation and confidence for each answer. The calculations of confidence were not used to halt the running of the algorithm.

Very intriguingly, the algorithm showed a clear instability in its ability to choose between either of the alternative interpretations for each set of the Necker problems tested. In fact, in no case was there any preference for either interpretation which was determined unambiguously, even though the confidence values for the interpretation exceeded that corresponding to a confidence of 95% for a sample set of two. The following charts plot the deviation of the interpretation similarity values against the level of abstraction, from coarsest to finest, for each of the sets, plainly showing the oscillation between interpretations.

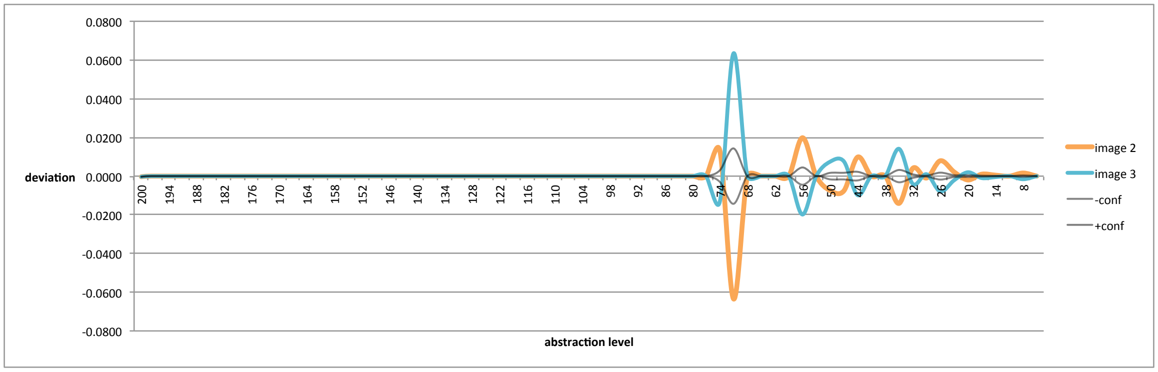


Figure 11.4. Deviation oscillations for Set 1.

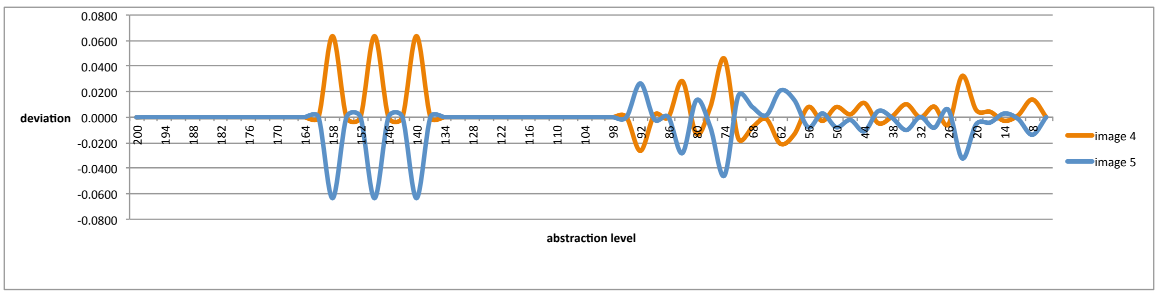


Figure 11.5. Deviation oscillations for Set 2.

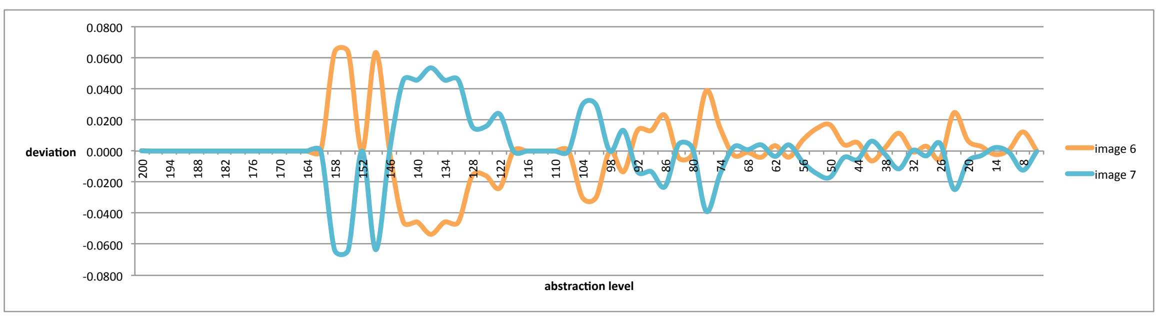


Figure 11.6. Deviation oscillations for Set 3.

There are occasional oscillations in the deviation at some coarse levels of abstraction, particularly apparent in Sets 2 and 3. Then, a regular pattern of oscillation appears to occur in each set after the abstraction level dips below 100x100. I attribute some of this to the manner in which the fractal representation is calculated: at a given partitioning level, an empty, temporary image buffer is calculated which is an even multiple of the partitioning in both directions, and then the image is composited into the center of that temporary buffer. The oscillations present in Sets 2 and 3 suggest this, but Set 1's chart does not. My interpretation in that case is that the haloed line effect used in Set 1 is not a remarkable feature within the image until the partitioning reaches a lower limit; thus, Set 1 deviations remain almost perfectly flat for much of the coarse abstractions.

An examination of the similarity values for each set is similarly revealing.

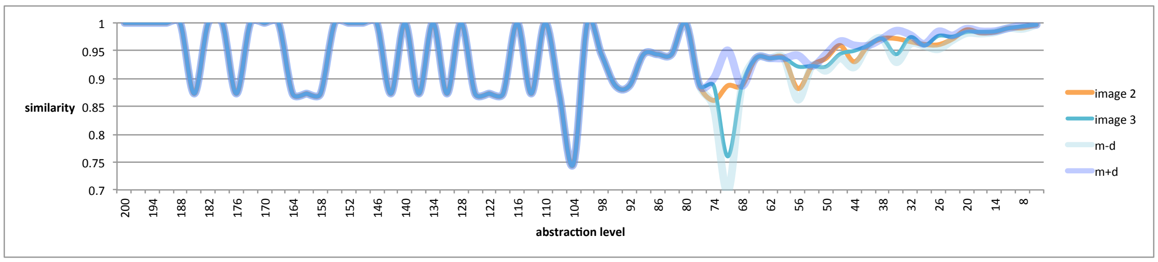


Figure 11.7. Similarity values for Set 1.

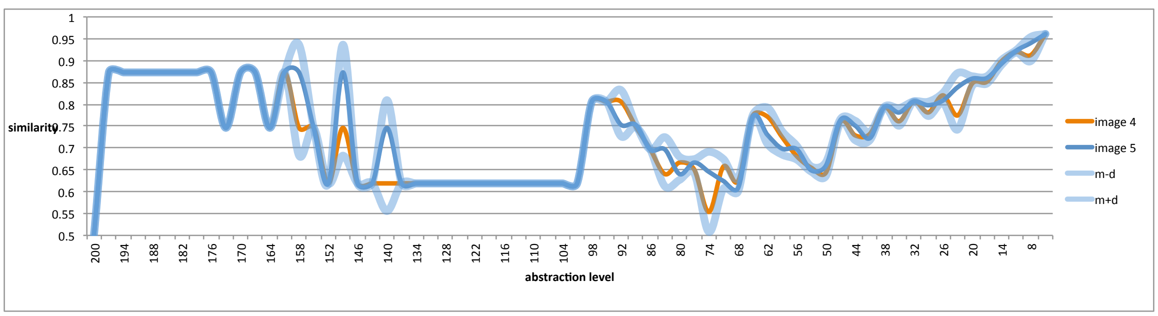


Figure 11.8. Similarity values for Set 2.

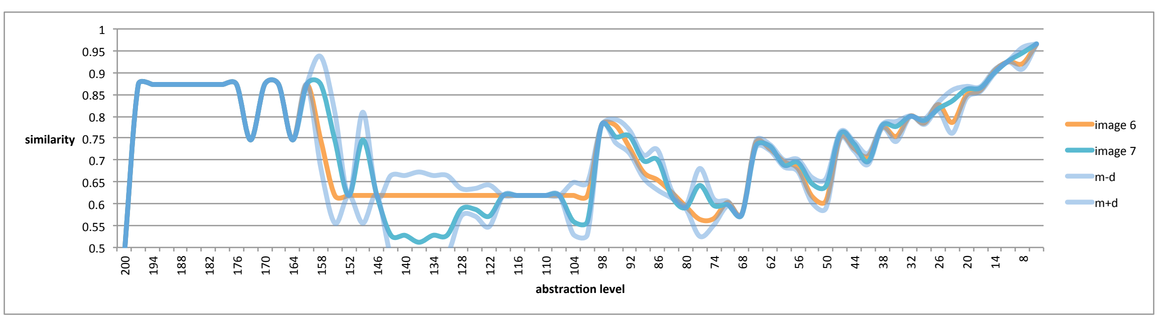


Figure 11.9. Similarity values for Set 3.

In these charts, it can be seen that the similarity value itself gyrates sporadically at coarse abstraction levels, and then, somewhere between 100x100 and 80x80, settles into a gradual rising pattern of oscillation about the mean, for each of the sets.

Unsurprisingly, the similarity value itself creeps upward to approach 1.0 as the abstraction level becomes ever finer. This is due to the increase in the homogeneity of the features being considered, coupled with the shear number of features under consideration (more than 200,000 at 5x5).

Implications

To my knowledge, this is the first computational model of the Necker cube which directly examines the bi-stable interpretation. Even when presented with sets of potential interpretations with varying visual cues, the model exhibits an inability to determine an unambiguous and significant interpretation of the Necker cube's orientation. This suggests that the analogical reasoning afforded by the fractal representation and illustrated via the ABR* algorithm (from which the Fractal Necker algorithm is derived) may offer some insight into the gestalt perceptual capabilities of humans.

As noted at the outset of this chapter, these results are very preliminary, and additional research will be needed to prove whether or not the techniques above are sufficiently robust for a variety of candidate interpretation images.

CHAPTER 12

FRACTALS AND IMPLICATIONS

All of the prior chapters in this dissertation concern themselves with either a motivation of some particular point of fractal representations or describe the results of experiments. In this chapter, I shall summarize the defense of my thesis, and discuss the contributions and implications of the research.

The Summary

My dissertation has presented a very specific model of reasoning, one rooted in the utility of fractal representations, as a means of addressing in a computationally feasible manner certain problems of visual similarity and visual oddity. I hold that my research is sufficient to permit me now to assert that the hypotheses I made at the outset of this dissertation are substantiated claims, and that the sum of these claims substantiates my thesis. I present now those claims, and that summation.

Claim 1: the Fractal Representation is a Knowledge Representation

I claim that the fractal representation is a knowledge representation.

Development and Motivation

In Chapter 2, Fractals and Representation, I gave the background and motivation for the need to represent the received visual world in a novel way, one that captured, as a core aspect, the inherent repetition and similarity at scale present in visual scenes. I described in detail the manner in which images may be encoded in a fractal manner. I illustrated how this fractal encoding was dependent upon not just the source and target images, but also upon the partitioning chosen, and how in this manner these are important initial conditions to which the encoding is sensitive. This led to four significant insights.

Insight 1: the Relationship between Source and Target

I described the insight that the relationship between the source and target images of the encoding could be considered as encoded as well in this fractal manner, even though in accordance with the Collage theorem the use of such a fractal encoding would necessitate convergence into the target image, regardless of source image. The specific insight was that the source image, in its role as an initial condition, significantly determined the encoded relationship. Furthermore, given the nature of the search at the heart of the encoding algorithm, which seeks similarity at different scales, I put forth that this encoding of similarity was a direct way to capture the visual analogical relationship between the two images.

Insight 2: the Partitioning as Abstraction “Knob”

As shown in chapter 2 and mentioned above, the partitioning of the images into subimages which are then used in the pattern-matching search at the heart of the fractal encoding algorithm can itself be considered as a significant initial condition. I showed that this partitioning decision determines the size of the encoding, the amount of time required to perform the overall encoding, and the fidelity with which the encoding captures the colorimetric information in the images. In this manner, the partitioning itself determines the level of abstraction of the information.

As it is a formal initial condition, however, the partitioning is seen as a useful "knob" with which to vary the information about the relationship between the source and target images. The advent of this "knob" precisely affords the ability to re-represent that relationship at a different level of abstraction should the need arise. Thus, the insight was that re-representation was so afforded, that subsequent re-representations could retain the same nature as the initial representation, and that the mechanism required for the re-representation was identical to the initial representation.

Insight 3: Features of Fractals

The fractal encoding algorithm, as described in Chapter 2, generates an encoding of the relationship between a source and a target image at a given level of abstraction. As the overall aim of my research addressed reasoning, the ability to judge a comparison between relationships of images became paramount. I noted that the set of transformations generated by the fractal encoding algorithm was unordered, providing a connotation of mutual independence between the transformations. Thus, I examined each transformation and developed a methodology of expressing each as a code consisting of a limited number of variables. Each of these variables, in turn, could be treated as labelled information (the color shift, the kind of similitude transformation used, and so forth).

In essence, each code I viewed as a Minsky-esque knowledge frame for that portion of the visual scene from which the transformation was derived, and the set of transformations in the fractal encoding then regarded as a frame-system. Furthermore, each code, having labelled variables, could be used, in whole or in subsets, as a means to index the encoding into memory. This interpretation in turn enabled the strategy for analogy making which I defend in the next section.

Insight 4: Mutuality

As my research progressed, the problem of addressing not just pairs of images, but groups of images, arose frequently. As an algorithm, the fractal encoding algorithm provided a means for generating a representation of the relationship between two images at some abstraction. However, I noticed that the fractal representation by itself had both a directionality (in the sense that a target image was considered in light of a source image) and an unordered set quality (in that the individual transformations that form the representation will cause convergence into the target no matter the order in which they are applied).

What was necessary, I realized, was that the representation must include not just the relationship from the source to the target, but also the relationship from the target to the source. The set-theoretic nature of the representation allowed me to describe therefore the mutual relationship between the images as the union of the two fractal representations, swapping the initial conditions of the images used in the encoding while maintaining the partitioning. Armed with this mutual representation, I extended it, in the manner as described in Chapter 2, to allow representations of three or more images, without losing any aspect of the encoded relationships.

From Fractal Encoding to Knowledge Representation

In Chapter 3, I argue that the fractal representation meets the criteria of a knowledge representation in several regards. I recap those arguments here briefly.

Fractal representations satisfy Markman's notion of a represented and representing world expressly through the particular initial conditions of the source and target images and the level of abstraction (as embodied by the partitioning), as via the commitment of inclusion/omission of that information into the representing world as given by the generated set of transformations. Markman's notion of representing rules find purchase in the fractal representation via the isomorphic mapping achieved by the partitioning scheme. The fractal representation is non-symbolic ala Markman in that it rests upon the non-arbitrary inherent structure determined by the representing rules, but I note that the manner in which the features extracted from the transformations of the representation may be regarded as symbolic. The expressivity and power Markman requires of a knowledge representation for the fractal representation rests in the extensibility of the mutuality I described above, the ability of the representation to represent any two (or more) arbitrarily chosen real-world images, and in the direct association of the representation to the mathematical notion of iterated function systems.

The fractal representation satisfies the roles required of a knowledge representation as given by Davis et al. (1990) as well. A fractal representation is a clear surrogate for two or more images, with a strong correspondence established by the manner in which the representation is achieved via the partitioning. As I argue in Chapter 3, this grounding correspondence is isomorphic and, in conjunction with the independence of the transformations under that partitioning, thereby makes an explicit, complete, and concise ontological commitment. As I show repeatedly through the dissertation, the fractal representation affords and sanctions a number of inferences, or perhaps better said, a number of ways in which the information contained within the representation may be combined and construed. Additionally, the various chapters of the dissertation illustrate precisely how the representation may be used for pragmatically efficient computation. Lastly, I argue in Chapter 3 that the fractal representation via its affordance of re-representation and featural similarity discovery offers a computational approximation of a medium of expression.

For all of the above reasons, I maintain that claim one is satisfied: the fractal representation is a knowledge representation.

Claim 2: the Identification of the Computational Strategy

I claim that using the fractal representation, a robust computational strategy may be determined which automatically adjusts the representation to an appropriate level of abstraction.

<u>The Extended Analogy By Recall (ABR*) Algorithm</u>

Regardless of which definition of analogy one chooses to adopt, all definitions require that some situation be regarded in comparison with another. That is to say, then, that two significant aspects of analogy making are that there is a comparison evaluated via some criteria and that that comparison involves one thing and another. Any strategy which would purport to address analogy making, it seems to me, then by necessity and at minimum would speak to both aspects: the manner of comparison to be conducted, and the manner by which the analog is chosen.

In the course of my research, and as I describe in detail in Chapter 4, I discovered, developed and implemented an original and novel algorithm, which I call the Extended Analogy By Recall (ABR*) algorithm. The ABR* algorithm addressed both aspects of analogy making I described above: it is based on the premise that analogy begins by being reminded of something (cf. Holyoak & Hummel, 2001), and therefore provides a mechanism for retrieving an analog, and it integrates the return of a measure of similarity along with that retrieved analog. The measure of similarity which the ABR* algorithm returns is based upon the commonality or rarity of the features found in the retrieved analog and the target, from which an original set of features are derived. The particular method of featural similarity which is used by the ABR* algorithm is based upon the theories and work of Amos Tversky (1977).

<u>Ambiguity, Confidence, and Abstraction-Shifting</u>

It is possible, even likely, that given some target, one or more analogs might be retrieved from memory using features derived from that target. In this case, the similarity metric which is returned along with the analogs may be used to distinguish which one of those analog should have priority (that is, which among them retrieved choices would be deemed to share the greatest featural similarity with the target). Unfortunately, the similarity measures for the analogs might be quite close in value, and although one might be numerically higher than the others, it is possible that it is not statistically significantly higher. Thus, one can readily see that there can arise a sense of ambiguity in two ways: a multiplicity of analogs, or a lack of a statistically significant singular analog. Thus, the algorithm would be able to offer only an ambiguous answer in response to some question, or lack the confidence with which to hold forth some particular answer.

In my work, I showed how the ABR* algorithm provides a way in which any ambiguity or uncertainty with which an answer to a visual analogy problem may be characterized can be attributed to those features naturally arising from fractal representations, as described above and in Chapter 2. I illustrated, in detail in Chapter 4, and in subsequent chapters and experiments, how the algorithm could use the advent of ambiguity as a means for triggering re-representation, using the fractal representation's inherent level-of-abstraction as a "knob." In this manner, I demonstrated how such a re-representation could be employed as a means of automatically adjusting the level of abstraction, successively moving through them, until a confident, unambiguous answer could be chosen. Subsequent chapters in the dissertation bear witness to the application of this automatic adjustment stratagem.

Thus, in Chapter 4, using as a visual similarity task as a basis, I presented a complete description of the ABR* algorithm, its motivation, and an argument that the reasoning embodied therein may be construed as a computational model of visual abstraction. Throughout the technical chapters of this dissertation, I illustrate regularly how the ABR* algorithm or algorithms directly derived from it are put into the service of solving visual analogy tasks. In this manner, I show that the strategy itself is robust across the several domains. Finally, as I developed the ABR* algorithm as a consequence of being inspired by aspects of human visual reasoning, and in particular, the human ability to shift the manner with which we regard some scene in order to facilitate understanding, I maintain that the algorithm on the whole is a computational strategy, yet cognitively-inspired.

For the reasons I have mentioned above, I maintain that claim two is satisfied: using the fractal representation, I have identified a robust computational strategy – the Extended Analogy By Recall (ABR*) algorithm – which automatically adjusts the representation to an appropriate level of abstraction.

Claim 3: the Utility of the Strategy for Visual Similarity Problems

I claim that using the fractal representation, a robust computational model can be derived for certain classes of problems of visual similarity, such as the Raven's Progressive Matrices tests.

<u>The work on Raven's Progressive Matrices</u>

In Chapter 5, I show the derivation of a new algorithm, Fractal Raven, based upon the ABR* algorithm. I use the Fractal Raven algorithm as a means to address all available problems from the entirety of those found in the Raven's Progressive Matrices suite. As my experiments show, the Fractal Raven algorithm detects the correct answer in 50 of the 60 problems of the Standard Progressive Matrices (SPM) test, 42 of the 48 problems on the Advanced Progressive Matrices (APM) test, 30 of the 36 problems on the Colored Progressive Matrices (CPM) test, and 50 of the 60 problems on the Standard Progressive Matrices Plus (SPM Plus) test. Insofar as I know, only one other computational model, that of my research colleague Maithilee Kunda, has been used against all available Raven's tests.

The results of Fractal Raven also illustrate an important aspect of my research, namely that although the answers were noted correctly, they were not always so noted uniquely or unambiguously. Indeed, the abstraction-adjustment strategy at the heart of the ABR* algorithm and present in Fractal Raven illustrated quite strongly that ambiguity and confidence are significant considerations and worthy of reporting for any computational model, yet for all prior models, none of them bring this into the discussion. Therefore, the Fractal Raven algorithm is the first computational model to address Raven's tests with confidence. Finally, although it was not the intention of my research to develop an algorithm which would demonstrate superior ability on the Raven's tests, as shown in Chapter 5 the results of Fractal Raven compare quite favorably to all other attempts.

<u>The work on Miller Analogies</u>

A characteristic of a Raven's visual similarity problem is that two or more analogical relationships must be preserved when selecting an answer. In other words, the problem is constrained in at least two ways. To illustrate the robustness of the overall strategy of ABR*, I additionally chose to experiment with the Miller Analogies test, and in particular, to draw upon examples first used by Evans in one of the first AI efforts on analogy making (Evans, 1964).

As I illustrated in Chapter 6, a Miller Analogies Test (MAT) problem consists of a single relationship, and the potential answer must maintain that analogous relationship. I derived an algorithm, Fractal Miller, based on the ABR* algorithm and as a direct descendant of the Fractal Raven algorithm, and conducted an experiment on all 20 problems used by Evans. The Fractal Miller algorithm detected the correct answer in 13 of the 20 problems, a score just slightly worse (13 vs. 15) than a contemporary computational model, as noted in Chapter 6.

<u>Summation for Claim 3</u>

For the reasons just mentioned, and as developed more fully in Chapters 5 and 6, I maintain that claim three is satisfied: I have successfully derived a robust computational model and employed it successfully against the entirety of the problems contained in the Raven's Progressive Matrices suite as well as those used classically by Evans in a test of visual Miller's Analogies.

Claim 4: the Utility of the Strategy for Visual Oddity Problems

I claim that using the fractal representation, a robust computational model can be derived for certain classes of problems of visual novelty, such as those found in the Odd One Out set. Herein, I support that claim.

On Visual Oddity

In Chapter 7, I discuss at length a problem of visual oddity, and particularly note the difficulty present in such problems: that the relationship between other objects in a scene are not known, nor are the number of those relationships known, and that all that is known is that some object or aspect of the scene is deemed odd. Immediately, this interposed a new aspect, that an object's relationship to all other objects in the scene, and not just some subset of pre-existing relationships, must be represented and considered. I developed, as a derivative of the ABR* algorithm, the Visual Oddity algorithm, and showed how oddity could be derived. Furthermore, I developed a means for distributing similarity measures to participating objects in relationships, to support the determination of oddity. Lastly, I showed that the unusual affordance of re-representation to differing levels of abstraction, using the fractal representation and confidence as described above, were useful and maintained in the Visual Oddity algorithm.

On the Odd One Out

In order to test the veracity of the Visual Oddity algorithm, I needed a set of problems, and found it in the work of Adam Hampshire and colleagues, in the form of almost 3,000 problems in their Odd One Out set, arranged in 20 levels of difficulty. Each problem in the Odd One Out set consisted of 9 images, arranged in a matrix fashion, and in each problem, there was exactly one image which did not belong with the rest – the so-called Odd One Out. No additional information was given.

In Chapter 8, I used the Visual Oddity algorithm to tackle the 2,976 problems of the Odd One Out. The Visual Oddity algorithm correctly identified the Odd One Out in an unambiguous fashion in 1,647 of the problems. The algorithm's performance was better on problems which were perceived as easier by human test takers than it was on problems perceived as more challenging by human test takers. Moreover, the results from this experiment also illustrated a significant aspect of the ABR* algorithm and the fractal representation: at coarse levels of abstraction, there are very few features over which to reason (tens to hundreds), whereas at very fine levels of abstraction, there are many, many more (hundreds of thousands). The scarcity of data at the coarse levels of abstraction led to mistakes, and at homogeneity of data at fine levels led to mistakes, as illustrated in Chapter 8. This new information, coupled with continued performance of the general abstraction shifting strategy, led to a refined version of ABR*, one in which the amount of data and the nature of that data also may be used as contributing factors in selecting an appropriate level of abstraction.

On Core Geometry

In another experiment, as explained in Chapter 9, I used a derivation of the Visual Oddity algorithm to address the problems used by Stanislaw Dehaene and colleagues to test whether humans have a naive understanding of certain geometric and mathematical concepts. The Dehaene test consisted of 45 visual oddity tasks, each containing six image of abstract geometric shapes. Five of the subimages in a Dehaene problem were related by some geometric or mathematical principle, such as alignment, chirality, or symmetry, but a sixth image was not. Thus, a Dehaene problem required the test taker to select the one that did not belong – a geometric rendition of an Odd One Out problem.

I derived a new algorithm, called CoreGeo, from the Visual Oddity algorithm, and applied it against the 45 Dehaene problems. As I report in Chapter 9, the CoreGeo algorithm detects a correct answer at a confidence of 95% or higher in 35 of those problems. In my analysis of the results, I discovered that the selection of a confidence threshold itself was not sufficient, and that the signal of an answer must be both unambiguous, confident, and strong. This is to say that the discovered oddity must be significant enough to warrant notice. I show in the chapter that a straightforward calculation of the coefficient of variation, a normalized measure of the dispersion of similarity values, is one way to determine such significance.

Summation for Claim 4

For the reasons summarized in this section, and developed fully in Chapters 7 through 9, I maintain that claim 4 is satisfied: using the fractal representation, a robust computational model has been derived for certain classes of problems of visual novelty, such as those found in the Odd One Out set.

Summation of the Defense

My research has shown, and this dissertation described in detail, the manner with which each of these claims have been satisfied:

- the fractal representation is a knowledge representation;
- using the fractal representation, a robust computational strategy may be determined which automatically adjusts the representation to an appropriate level of abstraction;
- using the fractal representation, a robust computational model can be derived for certain classes of problems of visual similarity, such as the Raven's Progressive Matrices tests; and
- using the fractal representation, a robust computational model can be derived for certain classes of problems of visual novelty, such as those found in the Odd One Out set.

The overall computational strategy which I developed, embodied in the Extended Analogy by Recall algorithm and fueled by reasoning over fractal representations, is novel in several senses: it is operating over fractal representations, which are themselves a new and novel contribution; it provides not only a retrieval of one or more source analogs from memory but a measure of the similarity associated with each; and it provides a parsimonious manner in which to shift or re-represent the elements over which it operates, based entirely upon the confidence and significance of the answer being return. The strategy is feasible, as I have demonstrated the ability to conduct various experimental runs using an instantiation of the algorithm written in the Java programming language running on conventionally available computer hardware. Lastly, the strategy is useful, for I have demonstrated precisely how it addresses problems in a variety of domains.

I therefore maintain strongly and confidently that my thesis statement is defended: reasoning using fractal representations is a novel, feasible and useful computational technique for solving certain problems of visual similarity and novelty.

The Contributions

My dissertation and the body of research it describes makes two primary, novel, and significant contributions to science. Those are the Extended Analogy by Recall (ABR*) algorithm and the fractal representation. Additionally, as I have noted in the preceding chapters, my research has other contributions which are of note. I shall now put each forward.

The Extended Analogy by Recall (ABR*) Algorithm

The first contribution is the Extended Analogy by Recall (ABR*) algorithm, a parsimonious, cognitively-inspired computational strategy for visual reasoning which automatically adjusts its representations to an appropriate level of abstraction. The several chapters of this dissertation show unmistakably that the strategy contained within the ABR* algorithm is suitable to meet the demands of a variety of visual analogy problems.

The Fractal Representation

The second contribution is the fractal representation itself, a new and novel knowledge representation that will open the door for analogy researchers, cognitive scientists, and computer scientists to explore the role self-similarity and perceptual complexity play in analogy making.

The Secondary Contributions

In addition to these primary contribution, several algorithms, which address reasoning specifically in visual similarity and visual oddity tasks, as well as algorithms which afford or mimic aspects of visual perception, are contributions in their own right.

<u>The Advent of Ambiguity Resolution</u>

In the course of developing the ABR* algorithm, and as a direct consequence of the fractal representation's affordance of re-representation via altering the initial condition of partitioning, I contributed a manner with which to shift the level of abstraction. By placing this shifting into service when and if the algorithm deems that an answer cannot be arrived at in an unambiguous and strong manner, this methodology of abstraction shifting becomes automated. Insofar as I am aware, no one has ever constructed or demonstrated such a parsimonious mechanism.

The Fractal Raven Algorithm

As I point out in Chapter 5, a significant amount of research has been devoted to the cognitive and computation study of the Raven's suite of visual similarity problems. My work on Raven's, and the Fractal Raven algorithm itself, contributes to and extends that research.

The Visual Oddity Algorithm

Visual oddity tasks, in the same manner as the visual similarity tasks such as the Raven's test, also command their fair share of research, both cognitively and computationally. My research in this area offers researchers a powerful new set of tools, the fractal representation and the regard of ambiguity resolution, as a means for delving into phenomena arising from their exploration.

The CoreGeo Algorithm

In Chapter 9, I remarked upon the work of Dehaene and others on whether humans possess innate mathematical or geometric reasoning or recognition skills. The CoreGeo algorithm, with its lineage to both fractal representations and the ABR* algorithm, provide both a way to consider and revisit observed effects in those experiments, as well as a means for categorizing and developing new problems which might elicit and discriminate finer effects.

The FractalNecker Algorithm

Although it is preliminary, to my knowledge, my research and the Fractal Necker algorithm provide the first computational model of the Necker cube which directly examines the bi-stable interpretation as a byproduct of confidence.

The Implications

I believe strongly that my research carries with it a number of indications and implications for artificial intelligence, cognitive science and vision. I believe this to be so because in no small part my efforts have been inspired by the various aspects of human visual reasoning and by my work and the work of others to transmute certain of those aspects into computational models.

Although I reserve the final chapter of this dissertation as the place for speculation about certain topics which may overlap these, I must share now, in this penultimate chapter, those indications that are specific consequences of the contributions. I also ask for a measure of forbearance, for any moderate attempt at suggesting implications calls for some degree of speculation, as may be noted here.

Implications for Perception

My research has begun with the receipt of some visual scene. As shown above and throughout this dissertation, what I sought to do was to provide as a surrogate for that arriving scene a fractal representation of it. From there, depending upon the task at hand–decide what is similar, decide what is novel, decide how to act–my research looked expressly to the representation, and to what it afforded and sanctioned. This is the way in which the algorithms I developed came about.

But what of the arriving visual input itself? As I mentioned, it is the observation of Mandelbrot (1982), and of Barnsley and Hurd (1992), that the world itself exhibits repetition and similarity at various scales. The deliberate choice of building a fractal representation from a fractal encoding of the arriving world scene at once grounds the representation in the world and yet abstracts all else of the world away, so that what remains is merely a recipe for how one might reconstruct the scene, iteratively. It is the arriving world, inbound with repetition and similarity at scale, which affords and sanctions the kinds of reasoning I discuss. The fractal representation packages this, concisely. This, then, would beg at once several implications for further exploration.

Firstly, let us suppose that we might construct scenes which do not exhibit these characteristics. Is it possible to do so? What would they be? Would we be able to say that they are analogous to nothing in our experience?

A particular implication worthy of future study would be that there may exist a continuum of visual images, not classifiable along the traditional means of color, spatial structure, etc., but along a fractal-like dimension. When reasoning from a fractal representation, my research focused only upon features derived from the various transformations at the core of the representation, and only upon each of them independently. Thus, as a first approximation, what might be gleaned from considering tandems or subsets of the transformations? Would the consideration of those subsets yield a measure of the overall scene complexity akin that entropy measure found by Ruiz (2009)?

Another characterization of the scene might well stem from an analysis of the visual noise present. While determining how "noisy" a scene is may be achieved by any number of computer vision methods, perhaps a characterization may be provided within the fractal representation itself. In doing so, a potential new measure may emerge: the likelihood that measures of similarity or oddity calculated from the representation will be sufficiently discriminable.

Yet another implication has to do expressly with the grounding of the representation in the scene itself. Let us suppose that although the scene is encoded in total, some aspects of the representation are omitted, accidentally or intentionally. Could this also form another technique for establishing the veracity of reasoning from the representation; that is, could the degree to which the representation is judged to be isomorphic with the received visual scene play a crucial role in subsequent calculations of certainty and significance? In precisely the same manner that it affords re-representation into finer or coarser abstraction, the fractal representation offers a direct means to judge that isomorphism in conjunction with the partitioning which is its initial condition. I imply that this is a worthy area of future study.

Two aspects of human neurophysiology also present opportunities. The orienting reflex, as a mechanism for studying that a subject has noted an anomaly, would seem an evident choice, and perhaps with careful attention to visual design, scenes with less or more self-similarity could be used in replication of those studies, with the intention of implicating the acts of encoding or the regard of complexity. Similarly, a study of human vision search, in the spirit of Treisman and Gelade (1980) but using scenes with known fractal complexity, as determined by a computational model of the scene from the fractal representation, might illuminate a distinction in the kinds of processing at play in vision at a glance versus vision with scrutiny. I should hope that there could be discovered a new event-related potential ala P3a (cf. Picton, 1992; Näätänen & Gaillard, 1983) which may be seen as variable in correlation with the fractal complexity of the regarded scene.

Finally, with regard to perception, the fractal representation and the work I have performed can be specifically characterized as viewer-centric regards of the arriving world. I have not pursued the extension of my techniques into the object-centric view. However, in the last chapter of this dissertation, I do discuss at some length the frameworks of David Marr, and offer some speculation about the utility of fractal representations in the service of object discovery.

Implications for Cognition

Beyond perception, once the fractal representation is arrived at, it must be put to use in service of some goal or task. In my research, those tasks varied from decision making concerning visual similarity or visual oddity to classifying simulated retinal regions in agents piloting a simulated world. What other tasks may be worth attention? How might the tasks require the representation itself to be arrived at in a different manner?

As I showed through the development of the ABR* algorithm, the analogy making begins with the selection of some source analog, and in particular retrieved from a prior perceptional history of potential analogs kept in a memory system in a fashion organized by the features derived from the fractal representation of those prior percepts. The fractal representation itself I have shown is sensitive to its triplet of initial conditions, the source image, the target image, and the chosen partitioning. It is possible to relax any of these initial conditions, and each yields interesting nuances.

Suppose that the partitioning itself is kept constant, and let us presuppose that it is impossible to regard the arriving visual scene at a different level of abstraction. This would correspond to receiving a scene at a glance. What if the need arose to regard the scene at a different level of abstraction? One method would be to infer coarser or finer abstractions for aspects of the representation. In the last chapter of this dissertation, in the section on fractal reasoning, I offer a very specific suggestion as to how this may be accomplished, and how fractal composition may be seen as inference.

Suppose that the target image itself is kept constant, but let us relax the constraint of choice of the source image. If this is so, then any available image may be used to take the place of the source, including that of the target image. If we should choose to take the target as the source, then the fractal representation would provide an interpretation of the visual scene in terms of itself. This then would yield a potential strongly significant signal as to the visual complexity, but this would only be knowable (by another process assessing the representation) if the representation carried information concerning its precise source and target.

In another stance, let us suppose that the source image is not provided, and that for some reason there is not a desire to encode the target in terms of itself. I suggest that another option exists, and that is to use as a source image other prior percepts from the perceptual history of the agent, reconstituted as necessary. Indeed, as I have shown via the derivation of the mutual fractal representation, any number of such percepts could be used as source images, and the lot could be combined to form a prior perceptually grounded fractal representation of the just arrived visual scene.

Such an interpretation holds substantial implications, on several fronts: the arriving visual scene could be said to be grounded in a consistent perceptual stream; the arriving visual scene could be viewed as determined by what was just prior "in mind"–a priming point of view; or the source image could be constructed from perceptual fragments either as the target image arrived, or brought to bear in a kind of mental imagery task. It is even possible to consider that an admixture of the two could be used: the arriving visual scene could be encoded first in terms of itself, and then, through the elicitation of source analogs from perceptual history, could be reinterpreted via re-representation into a fractal representation (or mutual fractal representation as outlined above). Any of these are worthy of future experimentation.

At present, there is no information carried into the fractal representation which indicates which if any of the aforementioned constraints are present or relaxed: the representation simply is what it is. This meta-knowledge concerning the construction of the fractal representation may be quite useful, as I outlined above, and could be at the discretion of some over-arching cognitive-like process.

The ABR* algorithm presumes that there is a desire on the part of the enacting agency to provide a confident answer to the task at hand. This drive may or may not actually exist in cognition, or it might be manifested at a differing priority than the primacy with which my research has regarded it. That the ABR* algorithm, using fractal representations, can generate not just a set of answers but a companion set of confidences (or rationales) for those answers should be useful to subsequent processing. The degree to which confidence may be a drive, and the manner in which it might be relaxed or augmented by other processes is worth exploration.

Some criticisms of published accounts of aspects of my research focus upon the fractal representation and ABR* algorithm's inability to account for why one answer is the answer to a problem. My response to those criticisms is straightforward: the why is always the same – the answer is shown to stand out via featural similarity in a statistically relevant manner. More intriguing is that this research points to a theory of how to decide not *why* something stands out, but *that* something stands out. This is deliberate, and driven by the design goal of noticing similarity and novelty. But, this does not mute nor does it diminish the request: a continued exploration of the fractal representation, the ABR* algorithm and its descendent techniques should focus on connecting the *that* to the *why,* from a knowledge-based point of view, and not merely a statistical one.

My research has specific implications for theories of analogy as well. Current theories of analogy approach regard the process of analogy-making as if it were a singular process of mind. In the current thinking, the process of analogy-making is based upon three core ideas: that analogy depends upon the capture of not just features but the relationships between objects or concepts (Holyoak & Thagard, 1996); that propositional representations (and not imagistic representations) are crucial for the concise expression of those transformations and relationships; and that the fit of an analogy depends upon the alignment between the structure of representations, not necessarily the content of knowledge represented (Gentner, 1983). In contrast, our research group has a long history of building an alternative account of analogy (Goel, 1997; Griffith et al., 2000; Davies et al. 2005). In our group's view, analogy is composed of multiple processes, some of which are based on just features (e.g. Kunda et al., 2013) and others capturing relationships (e.g. Yaner & Goel, 2007, 2008), some more focused on propositional representations (e.g. Davies & Goel, 2001) and others on imagistic representations (e.g. Kunda et al., 2010), and some intent on exploiting the organization of knowledge into abstraction hierarchies (e.g. Goel & Bhatta, 2004; Davies et al., 2009) while others examine the content of knowledge at specific abstraction levels (e.g. McGreggor et al. 2012). My research clearly illustrates a connectivity to this history, and shows that the fractal representation as an imagistic representation and the ABR* algorithm as a featural analogical process offer one example of a content-based theory of analogy.

Implications for Artificial Intelligence

I have developed a novel visual representation, the fractal representation, and shown the power of a computational model based upon reasoning afforded by it through successive derivations of the ABR* algorithm. I believe, however, that more specific development can be done in the arena of artificial intelligence via these tools.

In my opinion, the field of AI suffers through representational swings, and it is the reliance on kinds of representations (lately symbolic versus neuronal/nodal) that hold large sway upon it. My hope is that with the advent of the fractal representation, as a novel form which sits not quite cleanly in either the "good-old-fashioned AI" or machine-learning camp, that additional techniques and representations may be derived. Thus, my hope is that just via existence, the fractal representation and the work represented here be a catalyst.

I mentioned briefly above that the fractal representation may be viewed as a frame system, and that each of the attendant portions of the representation viewed as frames, expressly in the sense that Minsky first proposed them (Minsky, 1975). I want to underscore the significance of this mapping, now, from the point of view of artificial intelligence.

Too infrequently are the representations that some AI system uses actually grounded in the world. The outcome of such loose grounding is often that the system itself can be seen as brittle, focused upon a particular domain or world, and unable to transcend that domain to general utility.

I propose the fractal representation has an additional opportunity for AI, as one such example of a wholly grounded representation. Given that it is demonstrably grounded, however, would this be a sufficient condition to admit a computational model that does not suffer from brittleness?

It is for this reason that I suggest exploration in AI concerning not just the fractal representation, but of the ABR* algorithm as well. Insofar as it is now conceived, the ABR* algorithm depends not quite entirely upon the fractal representation per se, but upon the ability of a representation to afford re-representation in service of the task (and in the express case of ABR*, to reduce ambiguity). What other kinds of representations await discovery, that afford parsimonious re-representation and substantive grounding? Could representational families exist, in which re-representation swaps between kinds of representations, and yet stay within the familial set? Could the notions of grounding or re-representation be added to already familiar knowledge representations, and then those new derivations be pressed into the task of analogy making?

Finally, the fractal representation offers a tantalizing hint concerning the nature and advent of a knowledge production rule. The fractal representation takes a source image, a target image, and a partitioning as initial conditions. It is but a small stretch of viewpoint to consider it as taking some initial state (the source image), some final state (the target image), and a system of constraints (the partitioning), and in this manner describing or representing the transformation of the initial state into the final state. What does the fractal representation, with its reliance on feature space searching at the heart of its encoding process, say with respect to its utility as a production rule? What does the composition of mutual fractals involving several images suggest about the composition and chaining of rules? Does the ability to reason analogically about fractal representations offer a means for identifying similarity between rules?

This, perhaps above all the rest of the implications, might prove the most worthy to pursue for AI.

CHAPTER 13

FRACTALS AND FUTURE

All of the prior chapters in this dissertation concern themselves with either a defense of some point of my thesis or describe the results of experiments. This chapter is a departure, in that I intend to bring up ideas that occurred to me during the course of my research, some well-formed, others less so, and to speculate briefly on what the future may hold for fractal reasoning. I'll segment my remarks into two primary areas: those concerning the fractal representation, and those concerning fractal reasoning. Finally, I'll close with some general commentary.

Forward the Fractal Representation

The fractal representation was described in substantial detail in Chapter 3. Yet, as I worked on its development, there were several aspects of representations and fractal representations which came to mind. Some of those ideas, notably the ability to vary the grid size to facilitate re-representation at different levels of abstraction, found their way into the main body of my research. Others, though, remain to be explored. Here are some of those ideas.

The Image, Revisited

The images that I've used throughout my research and presented within this dissertation are two-dimensional arrays of pixel values, generally in the RGB color space. It is from these images that fractal representations are calculated, by forming some partition based on a desired level of abstraction, and then proceeding with reasoning. it is true that there are any number of photometric manipulations permissible on the image, some of them perhaps advantageous (conversion to grayscale, for example). Generally, regardless of the photometric manipulation, we put pixels in, and get pixels out. But, treating images as arrays of pixels is not the only manner in which a fractal representation may be constructed. The principle requirement of constructing the fractal representation is that the source and target images being represented are searchable.

One such image representation which meets the requirement of searchability and therefore would be suitable as input into the fractal representation is the Discrete Cosine Transformation (DCT) (Ahmed et al., 1974; Chen et al., 1977). A discrete cosine transformation is a finite set of data points (in this case, pixels) in terms of a series of cosine functions of varying frequencies. DCTs are used in the JPG and other images formats, due to their lossy compression characteristics. A strong recommendation in favor of using DCTs when comparing images is that the transformation into the frequency domain for blocks within an image capture compactly a sense of the texture within that block.

To prepare a source and a target image for fractal representation at a given level of abstraction (call it N for discussion purposes) using DCTs, one would first calculate DCT(target,N). This means that the entire target image would be re-represented as a set of DCT blocks, each derived from a NxN block taken in a regular fashion from the target image. Similarly, but distinctly, one would also calculate a set of DCT blocks of size NxN from the source image, but instead of forming a regular partition, one would calculate the set of ALL possible DCT blocks of size NxN. Then, given these two representations, DCT(target,N) and ALLDCT(source,N), a set of fractal codes may be calculated for each block in DCT(target,N), found by searching for the best matching DCT block in ALLDCT(source,N).

Interestingly, since the DCT blocks would contain frequency information, and not photometric information, the number of possible features per fractal code increases from a single photometric value (the shift for all pixels within the block) to a frequency shift vector (potentially as long as NxN, but generally much shorter, involving only the lowest frequency or two). The DCT block is arranged such that the first value represents the average overall photometric value of the block, and successive values represent higher frequencies. Searching for blocks becomes much faster, since the average value is compared first (comparing the average of a target block to the average of a source block).

Exploring the use of DCT-based fractal representations would be an interesting future angle of research.

The Eight Transformations, Revisited

One of the areas where the fractal representation could be extended is in its use of the eight similitude transformations. In the discussion of the fractal representation, I made the case that those eight were sufficient, given that the smallest unit of an image which could encode symmetry, etc., was a 2x2 set of pixels. As seen throughout this dissertation, the level of abstraction is almost always something greater than this smallest, finest level (the pedantic finest level would be 1x1, or a single pixel, which I shall discuss shortly).

One can suppose that it is possible to allow for any arbitrary rotation to be used, and along with it reflections across arbitrary axes. The consequences to the representation would be felt in two ways.

Firstly, the fractal code itself would have to be extended to include a representation of the angle of rotation of either the image or the axis of reflection or both. One could imagine such a representation could be of the form: $r\theta$ or $a\theta$, to represent a rotation of θ degrees or a reflection about an axis at θ degrees, respectively. Doing so would offer the opportunity to create new features from these representations as well, perhaps affording a finer degree of analogical comparison.

Secondly, and most impactfully, the runtime complexity of creating the fractal representation would increase substantially. Instead of searching for 8 possible transformations, the algorithm would now have to evaluate 2N additional matches per block in the partitioning scheme, where $N = |\{\theta_1, \theta_2, \ldots\}|$ the magnitude of all potential angles of rotation or reflection.

I believe that affording this flexibility to the fractal representation in no way would diminish it, and would provide potentially improved reasoning on the Dehaene set of geometry problems. It also would be likely to provide improved performance on several of the Ravens test suite problems.

Variations of the Fractal Code

I've just mentioned allowing a number of new transformations into the fractal code, but I can suggest a few other ideas as well.

<u>Coordinate system inferences</u>

The coordinate system used for images throughout this dissertation has been the Cartesian coordinate system, with the origin located at the upper leftmost extent of an image. However, there exists other ways in which to address pixels within the scene, or to interpret their location afterward.

One such system is the polar coordinate system (Korn & Korn, 2000). In it, the traditional (x,y) coordinate is transformed into a radius and angle pair (r,θ) indicating distance and direction from the origin. However, if one merely used the upper leftmost extent as the origin, this would yield values of θ between 0.0 and $\frac{1}{2}\pi$. If one instead locate the origin at the center of the image (e.g. at $(\frac{1}{2}w,\frac{1}{2}h)$, presuming an image that is wxh pixels in extent), then the value of θ would range from 0.0 to 2π. While this may be of interest, it does carry with it an additional constraint, one that is absent in the fractal code as presented, and that is that the image's extent is known. This may not always be the case, nor may it always be desirable. Thus, if one is restricted to only constructing an inference based upon the information already present in a fractal code, then the quadrant restriction for θ must remain, and the distance value can be readily calculated as the square root of the squares of the (x,y) coordinate. Even so, this would add distance and direction as features over which to calculate similarities.

Colorimetric inferences

The pixel shift value encoded in a fractal code, like its spatial coordinate brethren, can also be interpreted in a number of ways. If the average value of the whole image is known, then the shift value would be recast in terms of relative value to that average. Thus, if an image were predominantly light colored, a darker pixel shift value might have more value which matching for similarity as a relative measure. But let me be clear, this would still presume that the overall average luminosity of an image be known at the fractal code level, perhaps an undesirable requirement.

Another interpretation of the shift value, independent of the average image value, would be to change the scale of the value from a linear scale to a logarithmic scale (Hunt & Pointer, 2011). Thus, small changes would match small changes more crisply.

A third possible way to conduct the interpretation would be to discretize the values into some limited set of values, and therefore clump the matching potential for features which composite the shift value (Hall, 1989).

It is also possible to have these additional varying scales merely be rendered as additional features, and therefore move from one to three or more colorimetric features portions.

The Finest Abstraction

At the finest level of abstraction, a portion of an image is reduced to a single, orientation-free, photometric value, a pixel. Curiously, the fractal representation readily may represent a single pixel, as a colorimetric shift, an identity transformation, and some offset. Thus, the fractal representation can represent any arriving image at any desired level of abstraction.

A potential future experiment would be to allow the visual similarity algorithm to proceed down to this finest level, and note the coefficient of variation in the answers.

Probabilistic Coding

The search for a code yields a number of possible candidates, some of which are identically suited. My current implementation uses a heuristic of seeking the closest matching blocks which are spatially near the same location in the source and target images. However, these other, unchosen matches are known to the algorithm, yet discarded.

A variation of the fractal code, then, would be to introduce a notion of probabilistic coding. That is, the fractal code could make note of how many blocks were close (within some controllable δ) matches, and have that be a new feature aspect. This would offer a chance to reason over which blocks were most or least frequent in the source, a nice thought if the source image contains a high degree of self-similarity.

The Fractal Modality

All of this dissertation research concerns itself with visual images. But the visual sense is not the only one over which one could reason in a fractal-like manner. Any sense modality which receives or transduces frequencies could have its input represented in a fractal form. Audial fractal reasoning is an area to explore!

On Fractal Reasoning

Analogical reasoning, in general, has been the province of reasoning over propositional or strongly symbolic representations. There have been exceptions, among them my research, and that of my colleague Maithilee Kunda. Much of my research has been inspired by my earliest exposures to AI, in working with Janet Kolodner on the preamble to case-based reasoning (Kolodner, 1982; Kolodner et al., 1982). Here, I want to bring attention to specifically what additional kinds of reasoning fractal (or fractal-like) representations afford, and expand on other intuitions arising from the advent of the ABR* algorithm.

Inference as Composition

As I described earlier, Davis et al. (1993) note that knowledge representations play five distinct, critical roles. Those roles are as a surrogate, as a set of ontological commitments, as a fragmentary theory of reasoning, as a medium for pragmatically efficient computation, and as a medium of human expression. For the moment, let me presume that I've succeeded in my defense of the fractal representation as a knowledge representation, and focus on its ontological commitments and its impact on reasoning.

Fractals, Ontologically

The fractal representation clearly makes a set of commitments that both define the extent of the representation's capture of the world and define the way that extent is expressed or embodied within the representation ontologically. As Davis et al. (1993) and Sowa (2000) note, the representational power lies in the correspondence of the representation to something in the world and in the constraints that that correspondence imposes. As I've shown, the fractal code contains some features which are spatial and some which are photometric. The act of creating a fractal representation depends upon a partitioning of the image into these fractal codes which is itself spatial. The correspondence between the codes and portions of the image are firmly established, but so to are the constraints: the representation is only about spatial and photometric information. There is no commitment to the construction of more commonplace geometric features (lines, enclosed areas, figure/ground, etc.) intrinsic in the fractal representation.

Inferences

Even though the theory of reasoning arising from a representation may be implicit, it can be discerned by considering three aspects: what the representation defines as inferencing, the set of inferences it allows, and the subset of those inferences which it recommends. I'll discuss what I mean by fractal inferencing momentarily, but let us first examine the nature of the allowed and the recommended inferences.

Allowed inferences are those inferences which can be made from available information. As a representation might arise in any number of ways, so too might the allowed inferences vary. As Davis, et al., point out, this flexibility is acknowledged so as to admit the legitimacy of the various approaches. Having this flexibility at its core provides a framework for re-representation. Indeed, much of my research hinged upon the fractal representation's facility for re-representation.

Clearly, the set of allowable inferences may become untenably large. A smaller, constrained subset of these inferences is necessary. Whether by specifying the constraints with which to select recommended inferences, or by providing them somewhat explicitly, some process or reasoning or insight must be at work to frame them. They also observe that much of the reasoning which informs recommended inferences has been provided by observation of human behavior. While there are many possible inferences which can be drawn from the fractal encoding, my preliminary thoughts on fractal inferencing, and subsequent discovery of the recommended inferences, stems from another role of knowledge representations: that they are a medium of efficient computation.

Motivating Composition

I noted earlier that the initial partitioning has a defining effect upon the fractal representation, in that each area partitioned results in a single fractal code. Over the course of my research, I have experimented with varying the partitioning by using block sizes of many sizes, even to the extent wherein the entire image was considered as a single block. These experiments lead me to an obvious finding: the time required to represent an image fractally increases as the partitioning varies. Intriguingly, however, the encoding process that underlies creating fractal representations did not increase uniformly with a decrease in partition size. In fact, I observed that the very largest or very smallest partitions took roughly the same amount of computation time, but that partitioning at sizes between these extremes took dramatically more computational resources.

The search for matching blocks within the source and target images while creating fractal representations is the cause almost all of the computation time. But, when faced with these experimental results, I began to speculate on how to leverage the notion of faster runtimes at the extreme of levels of abstraction. In particular, I had an “Aha!” moment: I wondered whether computational performance might improve if I composed coarser partitioning from the faster finer partitioning. This lead me to conjecture that composition might be a form of fractal inference, and to develop a composition algorithm for fractal representations, which I now present.

The Fractal Composition Algorithm

The Fractal Composition Algorithm is a way to ensure that all possible compositions have been made from a given encoding. It begins with the existing partitioning as expressed in the fractal representation, and stops when no further compositions can be made.

Given the set $T = \{T_1, T_2, T_3, ..., T_n\}$ is a fractal representation.
Let B represent the level of abstraction at which T is encoded.
Do:

- Let $n \leftarrow |T|$
- Construct $T' \subseteq T$ such that $T_i \in T'$ iff the block size of T_i is B.
- **For each fractal code $T_i \in T'$:**
  - Propose a composition K, based on T_i.
  - Search for 3 appropriate codes in T' which both satisfy the rules of composition and correspond to needed elements of K.
  - If the appropriate codes cannot be found, proposed composition K is invalid.
  - If the appropriate codes are found, the proposed composition K is valid.
  - If K is valid, form a new fractal code K' from K, and add it to the set T: $T \leftarrow T \cup \{ K' \}$
- Set $B \leftarrow 2*B$.

Repeat until $n = |T|$

Algorithm 13.1. Fractal Composition Algorithm

Proposing Composition

A subset of the fractal codes in the fractal representation T is selected, based solely upon a single feature: its block size. It is important to note that the selection of this subset based upon this one feature is non-arbitrary: I choose this feature expressly because I am seeking to optimize the calculation of a coarser partitioning based on block size. Were it desirable, other partitionings of the fractal codes could be determined by extracting this subset using any of the feature (or combination of features) within a fractal code.

Given the selection of such fractal codes, I can now address what kinds of compositions might be made. In the present case, I know that one of these fractal codes will be one of four participating codes in a larger composition, as illustrated in Figure X.

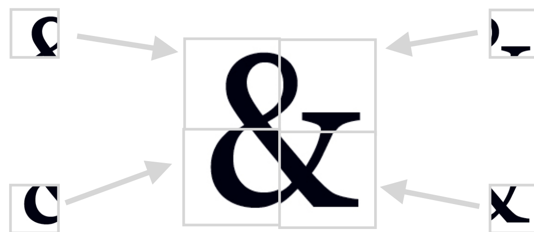
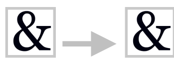
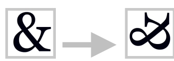
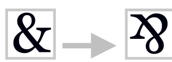
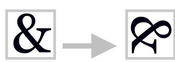
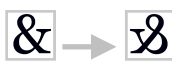
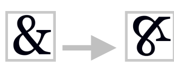
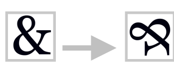
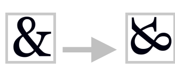


Figure 13.1. Image Composition

The affine transformation feature of the composed code I shall require, as a constraint in the present implementation, to be identical to the affine transformation associated with the given code. As the composition can be considered to be a 4-tuple of codes, this feature determines which position within that 4-tuple will be occupied by the given code. Table 13.1 illustrates the required tuple location, based upon the affine transformation.

Table 13.1. Initial location within the composed tuple.

Transformation	Name	Position	Tuple
	Identity	1	< C, *, *, * >
	Rotate 90°	3	< *, *, C, * >
	Rotate 180°	4	< *, *, *, C >
	Rotate 270°	2	< *, C, *, * >
	Flip Horizontal	2	< *, *, C, * >
	Flip Vertical	3	< *, *, C, * >
	Reflect XY	1	< C, *, *, * >
	Reflect -XY	4	< *, *, *, C >

Since each fractal code has a single affine transformation associated with it, and because I am constraining the composed code to have the same affine transformation, there will be exactly one proposed composition per selected fractal code. I note, however, that a code may participate in more than one composition.

The act of composition becomes searching for three other codes which can be combined to fill out the missing places in the 4-tuple and thus form the composition. While they are selected from the same subset of fractal codes as the original code, they are further constrained by four rules, which I have labeled the rules of fractal composition.

Rules of Fractal Composition

I believe that these four rules are minimally necessary to ensure candidacy for combination. While the order of application of these rules is generally arbitrary, computational efficiency may be gained by pruning the subset of codes through a particular ordering (size, transform, photometric) prior to calculation of coverage. The rules subdivide into two groups: the consistency rules, and the coverage rule.

Consistency Rules: the Photometric, Size, and Transform Rules

A fractal code contains both spatial and photometric features. The photometric rule holds that the photometric features of the codes being combined must be the same. Thus, the pixel operation and the amount of color shift must be the same across all candidate codes. The size rule requires that the dimensions of the code being composed be exactly twice that of the constituent code. The algorithm directly enforces this rule during the selection by block size of the subset of codes to consider. Lastly, the transform rule constrains the candidate codes to possess the same affine transformation as that of the original code.

The Coverage Rule

The codes being combined must be spatially adjacent to one another and completely cover the area under consideration. However, they must not overlap. I use region connection calculus (reference) to specify how each of the regions must be externally connected (in that they share borders but do not overlap). Furthermore, these regions must exhibit this connectivity in the destination image space. Thus, for a proposed combination 4-tuple $<C_1,C_2,C_3,C_4>$ and their corresponding regions $\{R_1,R_2,R_3,R_4\}$, the following four RCC relationships must hold:

$$\mathbf{R_1 \text{ EC } R_2 \wedge R_3 \text{ EC } R_4 \wedge R_1 \text{ EC } R_3 \wedge R_2 \text{ EC } R_4}$$

where, given point-set closed regions A and B,

$$\mathbf{A \text{ EC } B \coloneqq A \cap B \neq \varnothing \wedge \text{interior}(A) \cap \text{interior}(B) = \varnothing}$$

$$\mathbf{\text{interior}(A) \coloneqq \text{the set of interior points of } A}$$

Confirming Composition

Once candidate blocks are found which conform to the rules of composition, then the task is to confirm the composition. This can be done quite readily by examining the proposed composition under the coverage rule, but instead of using the destination image space, I use the source image space. If all four of the RCC relationships hold in source image space, then the proposed composition is valid.

A new fractal code may be generated from the proposed composition by taking the source and offset spatial values from the first value of the 4-tuple, and using the twice the block size, the affine transformation, and the color shift which were constant across all constituents.

Is Composition Truly Inference?

The result of the composition algorithm is the creation of additional fractal codes, which are added to the entire representation. Can it be said that these additional codes are the result of inference?

Preliminarily, I'd argue that yes, composition in this manner is inference. New knowledge is being created in a manner wholly sanctioned by the representation. Additionally, that new knowledge is then kept within the same representational framework as the old knowledge. From a fractal reasoning point of view, this means that there would now be available, for any given initial level of abstraction, a set of inferred fractal codes, each of which would contribute their own fractal features over which analogy by recall could occur.

Visual Case-based Reasoning

Case-based reasoning concerns itself with performing analogical reasoning over cases stored in memory (Kolodner, 1993; Riesbeck & Schank, 1989). My ABR algorithm was entirely informed by this field. Yet, in the ABR algorithm, and not so with case-based reasoning, I make use of how something is stored in memory, rather than what something is stored in memory. The use of features as indices into memory aligns with case-based reasoning, but differs in that no quality assessment is made with respect to the index itself. Every index is treated equally.

One could make an extension to my algorithms and models by relaxing this equality among indices. After all, each index for me is a feature vector of some degree. It is easy to see how a weighting system could be implemented (e.g. the more features have that particular index, the stronger that index becomes when calculating similarities). Likewise, it is easy to see how to extend the comparison of indices, by relaxing the binary nature of the current comparison of the feature (to wit, in the present implementation, my algorithms view feature matching as all-or-nothing). Proximal feature matching, on a 0.0 to 1.0 scale, could yield a more finely tuned match, where the feature preference parameters could be determined a priori by the problem under consideration.

In such a light, the Analogy By Recall algorithm and its descendants can be considered as a form of visual case-based reasoning.

The long shadow of Marr

David Marr was a highly influential researcher in vision (Marr, 1982; Frisby & Stone, 2010). Famously, he proposed two frameworks for evaluating computational systems and for how the human visual system may work.

The first of these, his computational framework as outlined in Frisby & Stone (2010), informs and inspires my work as much as any other. In this framework, he proposes that computational systems be analysis at three distinct levels:

* the computational level, to identify the constraints for solving some problem: what is the nature of the problem to be solved, what i the goal of the computation, why is it appropriate, and what is the logic of the strategy by which it can be carried out?
* the representational and algorithmic level, in which the constraints are put to work in an algorithm: how can the computational theory be implemented, what representations are to be used, and what is the algorithm which transforms the input into the output?
* the hardware implementation, as the realization of the algorithm and the representation: what is the physical nature of the processing?

With this analytical approach, Marr did not mean to imply that the way the brain interprets visual information would be best seen as some series of steps, but rather that this affords a way to characterize the advent and realization of constraints when performing that interpretation. Marr also did not equate this framework with computer vision per se, and rather used it to lump problems into two "types." A "type one" problem would be one in which some computational account would be attainable in principle, and therefore subject to his three-level analysis. A "type two" problem would offer only evidence as to the interplay of various components, and not to the specifics of those components (and therefore not subject to the three-level analysis). It is safe to presume at this writing that how it is that the brain “sees” a visual scene is a task which we can ascribe to neither of Marr’s problem types.

Marr’s other framework forms a potential account of how the human visual system may see a scene. In Marr and Hildreth (1980), they describe how to form, first, a raw primal sketch of edges found within a presented visual scene, postulating that the first few stages of a brain’s visual processing system affords the neuronal apparatus necessary to detect edge fragments. Then, from these fragmentary raw edges, a full primal sketch may be determined, using principles of gestalt perception and reasoning (such as grouping, continuity, closure, and the principle of least commitment). From this slightly higher level representation, additional reasoning occurs to form planar closed shapes, and so on.

I offer a different thought: what if the material upon which Marr’s first stage of visual processing were not fragmentary edges, but fractal codes? My supposition here is that just as Marr leverages gestalt continuity to assist in joining edge fragments into full primal edges, reasoning from fractal features, based upon an agent’s prior perceptual history, could give rise to similar gestalt-like capabilities. Perhaps even such a system could operate in tandem with a Marr process, influencing and informing the inferences made at each of Marr’s stages.

Final Remarks

Fractal representations are analogical in that they have a structural correspondence with the images they represent. Like other knowledge representations, fractal representation support inference and composition. In this dissertation, I've used fractal representations to develop a powerful new model of analogical reasoning and applied it to problem domains of visual similarity and visual oddity. Along the way, I also illustrated its utility in providing a nascent perceptual capability for agents in a virtual world, and even demonstrated its ability to mimic bistable perception. The sum of my research suggests a degree of generality to fractal representations for addressing visual analogy problems.

Analogies are based on similarity and repetition. Fractals capture self-similarity and repetition within images at multiple scales. Thus, the fractal representation brings the powerful idea of self-similarity to analogy-making. Furthermore, since fractals work at multiple scales, they give to rise to an iterative problem solving strategy as I have demonstrated. Processing may begin at a certain level of abstraction for computational efficiency or other expediencies, but should the problem solving not result in a clear answer, the strategy may be shifted to other levels of abstraction.

All of this is a direct consequence of choosing to represent images fractally.

REFERENCES

Ahmed, Nasir, Natarajan, T, & Rao, Kamisetty R. (1974). Discrete cosine transform. Computers, IEEE Transactions on, 100(1), 90-93.

Allers, R., & Minkoff, R. (Writers). (1994). The Lion King: Walt Disney Pictures.

Appel, Arthur, Rohlf, F James, & Stein, Arthur J. (1979). The haloed line effect for hidden line elimination (Vol. 13): ACM.

Arkin, Ronald C. (1998). Behavior-based robotics [electronic resource]: MIT press.

Arnheim, Rudolf. (1954). Art and visual perception: A psychology of the creative eye: Univ of California Press.

Ashby, F Gregory, & Ennis, Daniel M. (2007). Similarity measures. Scholarpedia, 2(12), 4116.

Atmanspacher, H, Bach, M, Filk, T, Kornmeier, J, & Romer, H. (2008). Cognitive time scales in a Necker-Zeno model for bistable perception. Open Cybernetics & Systemics Journal, 2, 234-251.

Baliarda, Carles Puente, Romeu, Jordi, & Cardama, Angel. (2000). The Koch monopole: A small fractal antenna. Antennas and Propagation, IEEE Transactions on, 48(11), 1773-1781.

Barnsley, Michael, & Hurd, Lyman. (1992). Fractal Image Compression. Boston, MA: A.K. Peters.

Barnsley, Michael F, & Demko, Stephen. (1985). Iterated function systems and the global construction of fractals. Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, 399(1817), 243-275.

Barnsley, Michael F, & Sloan, Alan D. (1990). United States Patent No. 4,941,193.

Barsalou, Lawrence W. (1999). Perceptual symbol systems. Behavioral and brain sciences, 22(04), 577-660.

Barsalou, Lawrence W. (2008). Grounded cognition. Annu. Rev. Psychol., 59, 617-645.

Berger, Andrea, Henik, Avishai, & Rafal, Robert. (2005). Competition between endogenous and exogenous orienting of visual attention. Journal of Experimental Psychology: General, 134(2), 207.

Biederman, Irving. (1987). Recognition-by-components: A theory of human image understanding. Psychological review, 94(2), 115-147.

Bisiach, Edoardo, Ricci, Raffaella, Lai, Elena, De Tanti, Antonio, & Inzaghi, Maria Grazia. (1999). Unilateral neglect and disambiguation of the Necker cube. Brain, 122(1), 131-140.

Bobrow, Daniel G, & Winograd, Terry. (1977). An overview of KRL, a knowledge representation language. Cognitive science, 1(1), 3-46.

Bohan, A, & O'Donoghue, D. (2000). LUDI: A model for geometric analogies using attribute matching. Paper presented at the Proceedings of the 11th Artificial Intelligence and Cognitive Science Conference (AICS-2000), Galway, Ireland.

Bringsjord, Selmer, & Schimanski, Bettina. (2003). What is artificial intelligence? Psychometric AI as an answer. Paper presented at the International Joint Conference on Artificial Intelligence.

Brooks, Rodney A. (1991). Intelligence without representation. Artificial intelligence, 47(1), 139-159.

Bush, Robert R, & Mosteller, Frederick. (1953). A stochastic model with applications to learning. The Annals of Mathematical Statistics, 559-585.

Carpenter, Patricia A, Just, Marcel Adam, & Shell, Peter. (1990). What one intelligence test measures: A theoretical account of the processing in the Raven Progressive Matrices Test. Psychological review, 97, 404-431.

Cassimatis, Nicholas L, Bello, Paul, & Langley, Pat. (2008). Ability, Breadth, and Parsimony in Computational Models of Higher−Order Cognition. Cognitive science, 32(8), 1304-1322.

Cave, Kyle R, & Bichot, Narcisse P. (1999). Visuospatial attention: Beyond a spotlight model. Psychonomic Bulletin & Review, 6(2), 204-223.

Chen, Wen-Hsiung, Smith, CH, & Fralick, Sam. (1977). A fast computational algorithm for the discrete cosine transform. Communications, IEEE Transactions on, 25(9), 1004-1009.

Cirillo, Simone, & Ström, Victor. (2010). An Anthropomorphic Solver for Raven's Progressive Matrices.

Cross, Mark C, & Hohenberg, Pierre C. (1993). Pattern formation outside of equilibrium. Reviews of Modern Physics, 65(3), 851.

Dastani, Mehdi, & Indurkhya, Bipin. (1997). An algebraic approach to similarity and categorization. Paper presented at the Interdisciplinary Workshop On Similarity and Categorization.

Dastani, Mehdi, & Indurkhya, Bipin. (2000). Discovering Pattern Structures: Interaction between Analogical Reasoning and Perception.

Davies, Jim, & Goel, Ashok K. (2001). Visual analogy in problem solving. In Proc. of the 17th International Joint Conference on Artificial Intelligence (IJCAI-01), 372-382.

Davies, Jim, Goel, Ashok K, & Nersessian, Nancy J. (2009). A computational model of visual analogies in design. Cognitive Systems Research, 10(3), 204-215.

Davies, Jim, Goel, Ashok K, & Yaner, Patrick W. (2008). Proteus: Visuospatial analogy in problem-solving. Knowledge-Based Systems, 21(7), 636-654.

Davies, Jim, Nersessian, Nancy J, & Goel, Ashok K. (2005). Visual models in analogical problem solving. Foundations of Science, 10(1), 133-152.

Davis, Randall, Shrobe, Howard, & Szolovits, Peter. (1993). What is a knowledge representation? AI magazine, 14(1), 17.

Dehaene, Stanislas, & Cohen, Laurent. (1994). Dissociable mechanisms of subitizing and counting: neuropsychological evidence from simultanagnosic patients. Journal of Experimental Psychology: Human Perception and Performance, 20(5), 958.

Dehaene, Stanislas, & Cohen, Laurent. (1996). Towards an anatomical and functional model of number processing. Mathematical cognition, 1, 83-120.

Dehaene, Stanislas, Izard, Véronique, Pica, Pierre, & Spelke, Elizabeth. (2006). Core knowledge of geometry in an Amazonian indigene group. Science, 311(5759), 381-384.

Demko, Stephen, Hodges, Laurie, & Naylor, Bruce. (1985). Construction of fractal objects with iterated function systems. Paper presented at the ACM SIGGRAPH Computer Graphics.

Diascro, Matthew N, & Brody, Nathan. (1994). Odd-man-out and intelligence. Intelligence, 19(1), 79-92.

Dosher, Barbara Anne, Sperling, George, & Wurst, Stephen A. (1986). Tradeoffs between stereopsis and proximity luminance covariance as determinants of perceived 3D structure. Vision research, 26(6), 973-990.

Eckmann, J-P, & Ruelle, David. (1985). Ergodic theory of chaos and strange attractors. Reviews of Modern Physics, 57(3), 617.

Einhäuser, Wolfgang, Martin, Kevan AC, & König, Peter. (2004). Are switches in perception of the Necker cube related to eye position? European Journal of Neuroscience, 20(10), 2811-2818.

Eriksen, Charles W, & St. James, James D. (1986). Visual attention within and around the field of focal attention: A zoom lens model. Perception & Psychophysics, 40(4), 225-240.

Evans, Thomas G. (1964). A heuristic program to solve geometric-analogy problems. Paper presented at the Proceedings of the April 21-23, 1964, spring joint computer conference.

Falkenhainer, Brian, Forbus, Kenneth D, & Gentner, Dedre. (1986). The structure-mapping engine: Department of Computer Science, University of Illinois at Urbana-Champaign.

Ferguson, Ronald W. (1994). MAGI: Analogy-based encoding using regularity and symmetry. Paper presented at the Proceedings of the sixteenth annual conference of the Cognitive Science Society.

Forbus, Kenneth D, Ferguson, Ronald W, & Usher, Jeffery M. (2001). Towards a computational model of sketching. Paper presented at the Proceedings of the 6th international conference on Intelligent user interfaces.

Forbus, Kenneth D, Usher, Jeffrey M, Lovett, Andrew, Lockwood, Kate, & Wetzel, Jon. (2008). CogSketch: Open-domain Sketch Understanding for Cognitive Science Research and for Education. Paper presented at the SBM.

Franco, Laura, & Sperry, RW. (1977). Hemisphere lateralization for cognitive processing of geometry. Neuropsychologia, 15(1), 107-114.

French, Robert M. (2002). The computational modeling of analogy-making. Trends in cognitive Sciences, 6(5), 200-205.

Frisby, John P, & Stone, James V. (2010). Seeing: The computational approach to biological vision: The MIT Press.

Gentner, Dedre. (1983). Structure-mapping: A theoretical framework for analogy. Cognitive science, 7(2), 155-170.

Gentner, Dedre. (1989). The mechanisms of analogical learning. Similarity and analogical reasoning, 199, 241.

Gentner, Dedre, Holyoak, Keith James, & Kokinov, Boicho N. (2001). The analogical mind: Perspectives from cognitive science: The MIT Press.

Gentner, Dedre, & Loewenstein, Jeffrey. (2002). Relational language and relational thought. Language, literacy, and cognitive development: The development and consequences of symbolic communication, 87-120.

Goel, Ashok K. (1997). Design, analogy, and creativity. IEEE Intelligent Systems, Special Issue on Qualitative Reasoning, 12(3), 62-70.

Goel, Ashok K., & Bhatta, Sambasiva R. (2004). Use of design patterns in analogy-based design. Advanced Engineering Informatics, 18(2), 85-94.

Goldschmidt, Gabriela. (2001). Visual analogy: A strategy for design reasoning and learning. Design knowing and learning: Cognition in design education, 199-220.

Goldstein, E Bruce. (2013). Sensation and perception: Cengage Learning.

Grassberger, Peter, & Procaccia, Itamar. (1983). Characterization of strange attractors. Physical review letters, 50(5), 346-349.

Green, David Marvin, & Swets, John A. (1966). Signal detection theory and psychophysics (Vol. 1): Wiley New York.

Gregson, RAM. (1976). A comparative evaluation of seven similarity models. British Journal of Mathematical and Statistical Psychology, 29(2), 139-156.

Griffith, Todd. W., Nersessian, Nancy J., & Goel, Ashok. (2000). Function-follows-Form: Generative Modeling in Scientific Reasoning. In 22nd Annual Conference of the Cognitive Science Society, 196-201.

Guarino, N. (1995). Formal ontology, conceptual analysis and knowledge representation. International Journal of Human-Computer Studies, 43(5), 625-640.

Guckenheimer, John, & Holmes, Philip. (1983). Nonlinear oscillations, dynamical systems, and bifurcations of vector fields.

Hall, Roy. (1989). Illumination and color in computer generated imagery (Vol. 7): Springer-Verlag New York.

Halsey, Thomas C, Jensen, Mogens H, Kadanoff, Leo P, Procaccia, Itamar, & Shraiman, Boris I. (1986). Fractal measures and their singularities: the characterization of strange sets. Physical Review A, 33(2), 1141.

Hart, John C. (1996). Fractal image compression and recurrent iterated function systems. Computer Graphics and Applications, IEEE, 16(4), 25-33.

Haugland, John C. (1981). Mind Design: Philosophy, Psychology, and Artificial Intelligence: MIT Press.

Henderson, John M. (2003). Human gaze control during real-world scene perception. Trends in cognitive Sciences, 7(11), 498-504.

Hertzmann, Aaron, Jacobs, Charles E, Oliver, Nuria, Curless, Brian, & Salesin, David H. (2001). Image analogies. Paper presented at the Proceedings of the 28th annual conference on Computer graphics and interactive techniques.

Hofstadter, Douglas R. (2001). Analogy as the core of cognition. The analogical mind: Perspectives from cognitive science, 499-538.

Hofstadter, Douglas R. (2008). Fluid concepts and creative analogies: Computer models of the fundamental mechanisms of thought (F. A. R. Group Ed.). New York: Basic Books.

Holyoak, Keith J, & Hummel, John E. (2001). Toward an understanding of analogy within a biological symbol system. The analogical mind: Perspectives from cognitive science, 161-195.

Holyoak, Keith J, & Thagard, Paul. (1996). Mental leaps: Analogy in creative thought: MIT press.

Holyoak, Keith J, & Thagard, Paul. (1997). The analogical mind. American Psychologist, 52(1), 35.

Hunt, Earl. (1974). Quote the Raven? Nevermore. In L. W. Gregg (Ed.), Knowledge and Cognition (pp. 129-158). Hillsdale, NJ: Erlbaum.

Hunt, R Reed. (1995). The subtlety of distinctiveness: What von Restorff really did. Psychonomic Bulletin & Review, 2(1), 105-112.

Hunt, Robert William Gainer, & Pointer, Michael R. (2011). Measuring colour: John Wiley & Sons.

Indurkhya, Bipin. (1989). Modes of analogy Analogical and Inductive Inference (pp. 217-230): Springer.

Indurkhya, Bipin. (1998). On creation of features and change of representation. 認知科学 = Cognitive studies: bulletin of the Japanese Cognitive Science Society, 5(2), 43-56.

Inui, Toshio, Tanaka, Shigeki, Okada, Tomohisa, Nishizawa, Sadahiko, Katayama, Masahiro, & Konishi, Junji. (2000). Neural substrates for depth perception of the Necker cube; a functional magnetic resonance imaging study in human subjects. Neuroscience letters, 282(3), 145-148.

Jackson, Peter (Writer). (2003). Lord of the Rings: Return of the King: New Line Cinema.

Johnson-Laird, Philip Nicholas. (1983). Mental models: Towards a cognitive science of language, inference, and consciousness (Vol. 6): Harvard University Press.

Junge, Kenneth. (1977). Generalization of the Eisler-Ekman similarity function. Perceptual and Motor Skills, 45(2), 616-618.

Kim, Kyungnam, Chalidabhongse, Thanarat H, Harwood, David, & Davis, Larry. (2005). Real-time foreground–background segmentation using codebook model. Real-time imaging, 11(3), 172-185.

Kishiyama, Mark M, & Yonelinas, Andrew P. (2003). Novelty effects on recollection and familiarity in recognition memory. Memory & cognition, 31(7), 1045-1051.

Klink, PC, van Wezel, RJA, & van Ee, R. (2012). United we sense, divided we fail: context-driven perception of ambiguous visual stimuli. Philosophical Transactions of the Royal Society B: Biological Sciences, 367(1591), 932-941.

Kogo, Naoki, Galli, Alessandra, & Wagemans, Johan. (2011). Switching dynamics of border ownership: A stochastic model for bi-stable perception. Vision research, 51(18), 2085-2098.

Kokinov, Boicho, & Petrov, Alexander. (2001). Integrating memory and reasoning in analogy-making: The AMBR model. The Analogical Mind. Perspectives from Cognitive Science, Cambridge Mass.

Kolodner, Janet. (1993). Case-based reasoning. San Mateo, CA: Morgan Kaufmann.

Kolodner, J., Simpson, B., Sharpe, D., & McGreggor, K. (1982). Intelligent Fact Retrieval. ACM SIGART Bulletin, 79, 55-56.

Kolodner, Janet L. (1982). The Role of Experience in Development of Expertise. Paper presented at the AAAI.

Korn, Granino Arthur, & Korn, Theresa Marie. (2000). Mathematical handbook for scientists and engineers: definitions, theorems, and formulas for reference and review: Courier Dover Publications.

Kornmeier, Jürgen, & Bach, Michael. (2004). Early neural activity in Necker−cube reversal: Evidence for low−level processing of a gestalt phenomenon. Psychophysiology, 41(1), 1-8.

Kornmeier, Jürgen, & Bach, Michael. (2005). The Necker cube—An ambiguous figure disambiguated in early visual processing. Vision research, 45(8), 955-960.

Krapivsky, PL, & Ben-Naim, E. (1994). Multiscaling in stochastic fractals. Physics Letters A, 196(3), 168-172.

Kuehne, Sven, Forbus, Kenneth, Gentner, Dedre, & Quinn, Bryan. (2000). SEQL: Category learning as progressive abstraction using structure mapping. Paper presented at the Proceedings of the 22nd Annual Meeting of the Cognitive Science Society.

Kuncel, Nathan R, Hezlett, Sarah A, & Ones, Deniz S. (2004). Academic performance, career potential, creativity, and job performance: can one construct predict them all? Journal of personality and social psychology, 86(1), 148.

Kunda, Maithilee, McGreggor, Keith, & Goel, Ashok. (2010). Taking a look (literally!) at the Raven's intelligence test: Two visual solution strategies. Paper presented at the Proc. 32nd Annual Meeting of the Cognitive Science Society, Portland.

Kunda, Maithilee, McGreggor, Keith, & Goel, Ashok. (2011). Two Visual Strategies for Solving the Raven's Progressive Matrices Intelligence Test. Paper presented at the Twenty-Fifth AAAI Conference on Artificial Intelligence.

Kunda, Maithilee, McGreggor, Keith, & Goel, Ashok. (2012). Reasoning on the Raven's Advanced Progressive Matrices Test with Iconic Visual Representations. Paper presented at the 34th Annual Meeting of the Cognitive Science Society, Sapporo, Japan.

Kunda, Maithilee, McGreggor, Keith, & Goel, Ashok K. (2013). A computational model for solving problems from the Raven's Progressive Matrices intelligence test using iconic visual representations. Cognitive Systems Research, 22-23, 47-66.

Laird, John E, Newell, Allen, & Rosenbloom, Paul S. (1987). Soar: An architecture for general intelligence. Artificial intelligence, 33(1), 1-64.

Lakoff, George, & Núñez, Rafael E. (2000). Where mathematics comes from: How the embodied mind brings mathematics into being: Basic books.

Lindenmayer, Aristid. (1968). Mathematical models for cellular interactions in development I. Filaments with one-sided inputs. Journal of theoretical biology, 18(3), 280-299.

Lohman, David F. (2004). Aptitude for college: The importance of reasoning tests for minority admissions. Rethinking the SAT: The future of standardized testing in university admissions, 41-55.

Lovett, Andrew, Forbus, Kenneth, & Usher, Jeffrey. (2007). Analogy with qualitative spatial representations can simulate solving Raven's Progressive Matrices. Paper presented at the Proceedings of the 29th Annual Conference of the Cognitive Society, Nashville, Tennessee.

Lovett, Andrew, Forbus, Kenneth, & Usher, Jeffrey. (2010). A structure-mapping model of Raven's Progressive Matrices. Paper presented at the Proceedings of the 32nd Annual Conference of the Cognitive Science Society, Portland, Oregon.

Lovett, Andrew, Lockwood, Kate, & Forbus, Kenneth. (2008). Modeling cross-cultural performance on the visual oddity task Spatial Cognition VI. Learning, Reasoning, and Talking about Space (pp. 378-393): Springer.

Lovett, Andrew, Tomai, Emmett, Forbus, Kenneth, & Usher, Jeffrey. (2009). Solving Geometric Analogy Problems Through Two−Stage Analogical Mapping. Cognitive science, 33(7), 1192-1231.

Lumer, Erik D, Friston, Karl J, & Rees, Geraint. (1998). Neural correlates of perceptual rivalry in the human brain. Science, 280(5371), 1930-1934.

Mandelbrot, Benoit B. (1975). Stochastic models for the Earth's relief, the shape and the fractal dimension of the coastlines, and the number-area rule for islands. Proceedings of the National Academy of Sciences, 72(10), 3825-3828.

Mandelbrot, Benoit B. (1982). The fractal geometry of nature. San Francisco: W.H. Freeman.

Mandler, George, & Shebo, Billie J. (1982). Subitizing: an analysis of its component processes. Journal of Experimental Psychology: General, 111(1), 1.

Markman, Arthur B. (1999). Knowledge representation: Routledge.

Markman, Arthur B, & Gentner, Dedre. (1993). Splitting the differences: A structural alignment view of similarity. Journal of Memory and Language, 32(4), 517-535.

Markou, Markos, & Singh, Sameer. (2003). Novelty detection: a review—part 1: statistical approaches. Signal processing, 83(12), 2481-2497.

Markou, Markos, & Singh, Sameer. (2003). Novelty detection: a review—part 2:: neural network based approaches. Signal processing, 83(12), 2499-2521.

Marr, D. (1982). Vision: A Computational Investigation into the Human Representation and Processing of Visual Information. San Francisco, CA: WH Freeman and Co.

Marr, David, & Hildreth, Ellen. (1980). Theory of edge detection. Proceedings of the Royal Society of London. Series B. Biological Sciences, 207(1167), 187-217.

Mazel, David S, & Hayes, Monson H. (1992). Using iterated function systems to model discrete sequences. Signal Processing, IEEE Transactions on, 40(7), 1724-1734.

McGreggor, Keith, & Goel, Ashok. (2011). Finding the odd one out: a fractal analogical approach. Paper presented at the Proceedings of the 8th ACM conference on Creativity and cognition.

McGreggor, Keith, & Goel, Ashok. (2012). Fractal analogies for general intelligence Artificial General Intelligence (pp. 177-188): Springer.

McGreggor, Keith, Kunda, Maithilee, & Goel, Ashok. (2011). Fractal Analogies: Preliminary Results from the Raven's Test of Intelligence. Paper presented at the Proc. International Conference on Computational Creativity, Mexico City, Mexico, April.

McGreggor, Keith, Yerga, Christopher M, & Van Brink, David. (1999). USA Patent No. 5,963,201. U. S. P. a. T. Office.

Meagher, Don. (2006). Understanding Analogies.

Meng, Ming, & Tong, Frank. (2004). Can attention selectively bias bistable perception? Differences between binocular rivalry and ambiguous figures. Journal of vision, 4(7).

Minsky, Marvin. (1974). A framework for representing knowledge.

Minsky, Marvin. (1995). A framework for representing knowledge. Paper presented at the Computation & intelligence.

Mitchell, Jude F, Stoner, Gene R, & Reynolds, John H. (2004). Object-based attention determines dominance in binocular rivalry. Nature, 429(6990), 410-413.

Näätänen, R, & Gaillard, AWK. (1983). 5 The Orienting Reflex and the N2 Deflection of the Event-Related Potential (ERP). Advances in psychology, 10, 119-141.

Nagao, Natsuki, Nishimura, Haruhiko, & Matsui, Nobuyuki. (2000). A neural chaos model of multistable perception. Neural Processing Letters, 12(3), 267-276.

Necker, LA. (1832). On an apparent change of position in a drawing or engraved figure of a crystal. The London & Edinburgh Philosophical Magazine and Journal of Science, 1(Third Series, November), 329–337.

Nersessian, Nancy. (2008). Creating scientific concepts: The MIT Press.

Neto, Hugo Viera, & Nehmzow, Ulrich. (2007). Visual novelty detection with automatic scale selection. Robotics and Autonomous Systems, 55(9), 693-701.

Newell, Allen. (1994). Unified theories of cognition (Vol. 187): Harvard University Press.

Newell, Allen, Shaw, John Calman, & Simon, Herbert A. (1958). Elements of a theory of human problem solving. Psychological review, 65(3), 151.

Noest, AJ, Van Ee, R, Nijs, MM, & Van Wezel, RJA. (2007). Percept-choice sequences driven by interrupted ambiguous stimuli: A low-level neural model. Journal of vision, 7(8).

Norman, Donald A. (1993). Cognition in the head and in the world: An introduction to the special issue on situated action. Cognitive science, 17(1), 1-6.

O'Hara, Scott, & Indurkhya, Bipin. (1995). Adaptation and redescription in the context of geometric proportional analogies. Paper presented at the AAAI-95 Fall Symposium on Adaptation of Knowledge for Reuse.

O'reilly, Randall C, & Johnson, Mark H. (1994). Object recognition and sensitive periods: A computational analysis of visual imprinting. Neural Computation, 6(3), 357-389.

Oblinger, Kenneth D Forbus Dan. (1990). Making SME greedy and pragmatic. Paper presented at the Program of the Twelfth Annual Conference of the Cognitive Science Society: 25-28 July 1990, Cambridge, Massachusetts.

Ojha, Amitash, & Indurkhya, Bipin. (2009). Perceptual vs. conceptual similarities and creation of new features in visual metaphor: New Frontiers in Analogy Research, Sofia: New Bulgarian University Press.

Owen, Adrian M, Hampshire, Adam, Grahn, Jessica A, Stenton, Robert, Dajani, Said, Burns, Alistair S, . . . Ballard, Clive G. (2010). Putting brain training to the test. Nature, 465(7299), 775-778.

Parish, Yoav IH, & Müller, Pascal. (2001). Procedural modeling of cities. Paper presented at the Proceedings of the 28th annual conference on Computer graphics and interactive techniques.

Pearson. (2011). The Miller Analogies Test Study Guide

Picton, Terence W. (1992). The P300 wave of the human event-related potential. Journal of clinical neurophysiology, 9(4), 456-479.

Pomplun, Marc, Ritter, Helge, & Velichkovsky, Boris. (1996). Disambiguating complex visual information: Towards communication of personal views of a scene. PERCEPTION-LONDON-, 25, 931-948.

Posner, Michael I, Snyder, Charles R, & Davidson, Brian J. (1980). Attention and the detection of signals. Journal of Experimental Psychology: General, 109(2), 160.

Prade, Henri, & Richard, Gilles. (2011). Analogy-making for solving IQ tests: A logical view Case-Based Reasoning Research and Development (pp. 241-257): Springer.

Prade, Henri, & Richard, Gilles. (2013). Analogical Proportions and Multiple-Valued Logics Symbolic and Quantitative Approaches to Reasoning with Uncertainty (pp. 497-509): Springer.

Prusinkiewcz, Przemyslaw, & Sandness, Glen. (1988). Koch curves as attractors and repellers. Computer Graphics and Applications, IEEE, 8(6), 26-40.

Prusinkiewicz, Przemyslaw, & Hammel, Mark S. (1992). Escape-time visualization method for language-restricted iterated function systems Fractal Geometry and Computer Graphics (pp. 24-44): Springer.

Prusinkiewicz, Przemyslaw, Lindenmayer, Aristid, Hanan, James S, Fracchia, F David, Fowler, Deborah R, de Boer, Martin JM, & Mercer, Lynn. (1990). The algorithmic beauty of plants (Vol. 2): Springer-Verlag New York.

Ragni, Marco, Stahl, Philip, & Fangmeier, Thomas. (2011). Cognitive Complexity in Matrix Reasoning Tasks.

Rasmussen, Daniel, & Eliasmith, Chris. (2011). A neural model of rule generation in inductive reasoning. Topics in Cognitive Science, 3(1), 140-153.

Raven, J, Raven, J.C., & Court, J.H. (2003). Manual for Raven's progressive matrices and vocabulary scales. San Antonio, TX: Harcourt Assessment.

Ray, Siddheswar, & Turi, Rose H. (1999). Determination of number of clusters in k-means clustering and application in colour image segmentation. Paper presented at the Proceedings of the 4th international conference on advances in pattern recognition and digital techniques.

Reeke, George N, & Edelman, Gerald M. (1988). Real brains and artificial intelligence. Daedalus, 117(1), 143-173.

Reynolds, Craig W. (1987). Flocks, herds and schools: A distributed behavioral model. Paper presented at the ACM SIGGRAPH Computer Graphics.

Reynolds, Craig W. (1999). Steering behaviors for autonomous characters. Paper presented at the Game Developers Conference, San Jose, CA.

Riesbeck, Christopher K., & Schank, Roger C. (1989). Inside case-based reasoning. Hillsdale, NJ: Lawrence Erlbaum Associates.

Ruiz, Philippe E. (2009). Measuring fluid intelligence on a ratio scale: Evidence from nonverbal classification problems and information entropy. Behavior research methods, 41(2), 439-445.

Ruiz, Philippe E. (2011). Building and solving odd-one-out classification problems: A systematic approach. Intelligence, 39(5), 342-350.

Russell, Stuart Jonathan, Norvig, Peter, Davis, Ernest, Russell, Stuart Jonathan, & Russell, Stuart Jonathan. (2010). Artificial intelligence: a modern approach (Vol. 2): Prentice hall Englewood Cliffs.

Santini, Simone, & Jain, Ramesh. (1999). Similarity measures. Pattern analysis and machine intelligence, IEEE transactions on, 21(9), 871-883.

Schaller, Robert R. (1997). Moore's law: past, present and future. Spectrum, IEEE, 34(6), 52-59.

Schneider, Jens, & Westermann, Rüdiger. (2001). Towards Real-Time Visual Simulation of Water Surfaces. Paper presented at the VMV.

Schwering, Angela, Gust, Helmar, Kühnberger, Kai-Uwe, & Krumnack, Ulf. (2009). Solving geometric proportional analogies with the analogy model HDTP. Paper presented at the Annual Meeting of the Cognitive Science Society (CogSci 2009), Amsterdam, The Netherlands.

Schwering, Angela, Krumnack, Ulf, Kühnberger, Kai-Uwe, & Gust, Helmar. (2007). Using gestalt principles to compute analogies of geometric figures. Paper presented at the 19th Meeting of the Cognitive Science Society (CogSci07).

Schwering, Angela, Kühnberger, Kai-Uwe, Krumnack, Ulf, & Gust, Helmar. (2009). Spatial cognition of geometric figures in the context of proportional analogies Spatial Information Theory (pp. 18-35): Springer.

Sezgin, Mehmet. (2004). Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic imaging, 13(1), 146-168.

Sinclair, N, & Schiralli, M. (2003). A constructive response toWhere mathematics comes from'. Educational Studies in Mathematics, 52(1), 79-91.

Sjoberg, Lennart. (1975). Models of similarity and intensity. Psychological Bulletin, 82(2), 191.

Sjoberg, Lennart. (1977). Similarity and multidimensional ratio estimation with simultaneous qualitative and quantitative variation. Scandinavian Journal of Psychology, 18(1), 307-316.

Sjöberg, Lennart. (1971). Three models for the analysis of subjective ratios. Scandinavian Journal of Psychology, 12(1), 217-240.

Sjöberg, Lennart, & Thorslund, Christer. (1979). A classificatory theory of similarity. Psychological Research, 40(3), 223-247.

Sloman, Aaron. (1989). On designing a visual system (Towards a Gibsonian computational model of vision). Journal of Experimental & Theoretical Artificial Intelligence, 1(4), 289-337.

Sloman, Aaron. (2011). What is vision for, and how does it work?

Smith, Alvy Ray. (1984). Plants, fractals, and formal languages. Paper presented at the ACM SIGGRAPH Computer Graphics.

Sokolov, Evgeniy N. (1963). Higher nervous functions: The orienting reflex. Annual review of physiology, 25(1), 545-580.

Sowa, John F. (2000). Knowledge representation: logical, philosophical, and computational foundations (Vol. 13): MIT Press.

Spearman, Charles. (1904). " General Intelligence," Objectively Determined and Measured. The American Journal of Psychology, 15(2), 201-292.

Stark, Jaroslav. (1991). Iterated function systems as neural networks. Neural Networks, 4(5), 679-690.

Steinman, Robert M, Pizlo, Zygmunt, & Pizlo, Filip J. (2000). Phi is not beta, and why Wertheimer's discovery launched the Gestalt revolution. Vision research, 40(17), 2257-2264.

Sternberg, Robert J. (1977). Intelligence, information processing, and analogical reasoning: The componential analysis of human abilities: Lawrence Erlbaum.

Sternberg, Robert J. (1985). Beyond IQ: A triarchic theory of human intelligence: CUP Archive.

Stone, James V, & Harper, Nicol. (1999). Temporal constraints on visual learning: a computational model. PERCEPTION-LONDON-, 28, 1089-1104.

Sundareswara, Rashmi, & Schrater, Paul R. (2008). Perceptual multistability predicted by search model for Bayesian decisions. Journal of vision, 8(5).

Taylor, Richard, Brewster, David, & Phillips, Richard. (1832). The London and Edinburgh Philosophical Magazine and Journal of Science: Richard Taylor, Red Lion Court, Fleet Street, Printer to the University of London. Sold by Longman, Rees, Orme, Brown, Green, and Longman; Cadell Baldwin and Cradock; Sherwood, Gilbert, and Piper; Simpkin and Marshall; and S. Highley, London:--by Adam Black, Edinburgh; Smith and Son, Glasgow; Hodges and M'Arthur, Dublin; and GG Bennis, Paris.

Tomai, E, Forbus, K, & Usher, J. (2004). Qualitative spatial reasoning for geometric analogies. Paper presented at the Proceedings of the 18th International Qualitative Reasoning Workshop.

Tomai, Emmett, Lovett, Andrew, Forbus, Kenneth D, & Usher, Jeffrey. (2005). A structure mapping model for solving geometric analogy problems.

Tong, Frank, Meng, Ming, & Blake, Randolph. (2006). Neural bases of binocular rivalry. Trends in cognitive Sciences, 10(11), 502-511.

Treisman, Anne M, & Gelade, Garry. (1980). A feature-integration theory of attention. Cognitive psychology, 12(1), 97-136.

Trick, Lana M, & Pylyshyn, Zenon W. (1993). What enumeration studies can show us about spatial attention: evidence for limited capacity preattentive processing. Journal of Experimental Psychology: Human Perception and Performance, 19(2), 331.

Trick, Lana M, & Pylyshyn, Zenon W. (1994). Why are small and large numbers enumerated differently? A limited-capacity preattentive stage in vision. Psychological review, 101(1), 80.

Tu, Xiaoyuan, & Terzopoulos, Demetri. (1994). Artificial fishes: Physics, locomotion, perception, behavior. Paper presented at the Proceedings of the 21st annual conference on Computer graphics and interactive techniques (SIGGRAPH '94).

Tversky, Amos. (1977). Features of similarity. Psychological review, 84(4), 327-352.

Tversky, Barbara. (1993). Cognitive maps, cognitive collages, and spatial mental models Spatial Information Theory A Theoretical Basis for GIS (pp. 14-24): Springer.

Ullman, Shimon. (1995). Sequence seeking and counter streams: a computational model for bidirectional information flow in the visual cortex. Cerebral Cortex, 5(1), 1-11.

Varma, Sashank. (2011). Criteria for the Design and Evaluation of Cognitive Architectures. Cognitive science, 35(7), 1329-1351.

Viola, Paul, & Jones, Michael. (2001). Rapid object detection using a boosted cascade of simple features. Paper presented at the Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on.

Von Restorff, Hedwig. (1933). Über die wirkung von bereichsbildungen im spurenfeld. Psychologische Forschung, 18(1), 299-342.

Wagemans, Johan, Elder, James H, Kubovy, Michael, Palmer, Stephen E, Peterson, Mary A, Singh, Manish, & von der Heydt, Rüdiger. (2012). A century of Gestalt psychology in visual perception: I. Perceptual grouping and figure–ground organization.

Wagemans, Johan, Feldman, Jacob, Gepshtein, Sergei, Kimchi, Ruth, Pomerantz, James R, van der Helm, Peter A, & van Leeuwen, Cees. (2012). A century of Gestalt psychology in visual perception: II. Conceptual and theoretical foundations.

Wallace, William P. (1965). Review of the historical, empirical, and theoretical status of the von Restorff phenomenon. Psychological Bulletin, 63(6), 410.

Whitely, Susan E, & Schneider, Lisa M. (1981). Information structure for geometric analogies: A test theory approach. Applied Psychological Measurement, 5(3), 383-397.

Winograd, Terry. (1973). A procedural model of language understanding. Computer models of thought and language, 152, 186.

Yaner, P. W., & Goel, A. K. (2006). Visual analogies at multiple levels of abstraction. In Proc. of 29th Annual Meeting of the Cognitive Science Society.

Yaner, P. W., & Goel, A. K. (2008). Analogical recognition of shape and structure in design drawings. International Journal of AI in Engineering Design, Analysis and Manufacturing, Special Issue on Multimodal Design, 22(2), 117-128.

Zhu, Song Chun, & Yuille, Alan. (1996). Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. Pattern analysis and machine intelligence, IEEE transactions on, 18(9), 884-900.