Visual Analogy in Problem Solving

Jim Davies and Ashok K. Goel
College of Computing
Georgia Institute of Technology, Atlanta
jimmydavies@usa.net, *goel@cc.gatech.edu*

in Proceedings of the International Joint Conference on Artificial Intelligence 2001.

Abstract

Computational models of analogical problem solving have traditionally described source and target domains in terms of their causal structure. But psychological research shows that visual reasoning plays a part for many kinds of analogies. This paper describes a model that transfers a solution from a source analog to a new target problem using only visual knowledge represented symbolically. The knowledge representation is based on a language of primitive visual elements and transformations. We found that visual knowledge is sufficient for transfer, but that causal knowledge is needed to determine if the transferred solution is appropriate.

1 Introduction

The goal of this work is to examine the nature and role of visual representations and inferences in analogical reasoning, and especially in *analogical transfer*. Analogy involves learning about some *target analog* by transferring knowledge from a *source analog*. The process consists of several steps: *retrieval* is identifying a candidate source analog in memory; *mapping* is finding the best set of correspondences between components of the analogs; *transfer* is the application of knowledge from the source analog to the target analog; *evaluation* is determining if the target problem has been solved appropriately; *storage* is storing the target analog in memory for potential reuse.

Traditional conceptual and computational theories of analogy have focused primarily on causal knowledge and inferences (see [Holyoak and Thagard, 1997] [Bhatta and Goel, 1997] [Falkenhainer *et al.*, 1990] for examples). Psychological research, however, shows that visual reasoning often occurs in analogy [Holyoak and Thagard, 1997], [Pedone *et al.*, 1999] Some recent theories [Bhatta and Goel, 1997] [Griffith *et al.*, 2000] represent structural knowledge in addition to causal knowledge. Structural knowledge describes a system's physical composition but typically includes only the information directly relevant for analyzing the causal behaviors of the system. Structural knowledge might be thought of as a schematic that shows the components of the system and the connections among them but leaves out other visual information such as what a wire looks like, which side of a pump is up, etc.

We define visual representations as those that consist only of information relevant to how an image appears. Note that this definition of "visual" includes both high-level symbolic representations and low-level bitmap representations (which only represent the locations of points of light). We view visual and causal knowledge as lying on a spectrum, where one extreme has raw sensory data, (such as a *bitmap* image), and the other has highly interpreted and abstracted knowledge (e.g., teleological knowledge). Visual knowledge is closer to the perceptual, or *modal*, end of the spectrum, and causal knowledge is nearer to the *amodal* end. Causality can only be represented *implicitly* in a visual representation. In contrast to a bitmap image, the visual knowledge we use contains abstractions of objects and relations, and is thus represented symbolically.

Our hypothesis is that symbolically represented visual knowledge provides a level of abstraction at which two otherwise dissimilar domains may look more alike. For example, the concepts of an army on the march and a ray of radiation are quite different, but if both are represented as lines, it may facilitate analogical retrieval, mapping and transfer. We hypothesize that evaluation, on the other hand, requires explicit causal knowledge: simply because the path of the army and the ray look alike does not imply that the two behave similarly. Since other's work has begun to explore the use of visual knowledge for mapping, our work focuses on analogical transfer.

In this paper, we sketch an outline of our computational theory of visual analogical transfer for a class of problems in which the source analog contains a sequence of images (or can be analyzed in terms of an image sequence). This theory has been implemented in an operational computer program called Galatea. We illustrate the theory using the classical fortress/tumor problem [Duncker, 1926] as an example. This example was chosen because psychological data indicates that experimental participants used visual inferences in solving it [Holyoak and Thagard, 1997]. In this task, experimental participants first read a story about a problem-solving situation: A general with a large army wants to overthrow a dictator who lives in a fortress. All roads to the fortress are armed with mines that will go off if many people are on them at the same time. To solve this problem he breaks up his army into small groups and has them take different roads. The groups arrive at the same time and take the fortress. Then, the subjects are given a new problem: A patient needs radiation treatment on a tumor inside the body, but the radiation will harm the healthy tissue it reaches on the way in. Finally, the participants are asked to solve the tumor problem. The analogous solution is to target the tumor with low-level rays coming from different directions, and have them converge on the tumor.

2 Language and Processing

Our first task was to design a language to express visual analogs and the maps between them. Since the theory pertains to sequences of images, we needed both a vocabulary of primitive visual transformations that express changes between two consecutive images, and a vocabulary of primitive visual elements that enable the transformations. We designed a primitive visualization language, called *Privlan*, which consists of such primitive visual elements and primitive visual transformations. It can represent diagram-like images and changes to them. Like other computational visual analogy theories, ours represents images as networks of symbols. In Privlan symbolic images are called *simages*, to differentiate them from bitmap images.

Privels: Primitive Visual Elements

Each simage is composed of a collection of primitive visual elements, or *privels*. Table 1 shows a list of some privels.

Table 1

Privel name	**attributes**
generic-visual-element	location, size
line	start-point, end-point thickness, location
circle	location, size
box	location, height, width, orientation

Objects in the domain, like the fortress, are associated with a privel type. Each privel type has attributes associated with it. As shown in Table 1, *lines* have a *start-point*, an *end-point*, a *location* and a *thickness*. These attributes are not strongly-typed. For example, the *end-point* of a *line* could be a location such as the “center” of the image or some component of the image, like the fortress.

Privits: Primitive Visual Transformations

Privlan represents changes to images over time with an ordered series of simages in different states. Each simage in the sequence is connected to any simages before and after it with primitive visual transformations, or *privits*. Table 2 shows some examples of privits.

Table 2

Privit name	**arguments**
move	object, new-location
decompose	object, number-of-resultants
put-between	object, first-object, second-object
add-component	object

Each privit can take arguments. *Move*, for example, takes some *object* that it is moving, and a *new-location*. It changes the *object's* value for the *location* attribute to the *new-location*.

For example, imagine a circle moving from the top of the image to the bottom. Privlan would represent this as a series of two simages. The first simage would contain a *circle* with *location* set to top. The second would be to have another circle (called, say, *circle-1*) represented whose *location* would be set to bottom. Privlan knows these two simages are in a series because they are connected with a *transform-connection*, which in turn is associated with a series of correspondences between objects in the simages: There would be a map between the *circle* in the first simage and *circle-1* in the second. This map between the circles would be associated with the *move* privit.

2.1 Algorithm

The bottom series of simages in Figure 2 shows a representation of the solved fortress problem analog. The bottom left simage is the initial state of the problem. The top series of simages shows the target analog, the tumor problem. The darkly shaded box shows the output of the system. The first simage is all that is input of the tumor problem.

To make an analogical transfer, the source and target analogs must have an analogy between them. The analogy between the first tumor problem simage and the first fortress problem simage specifies maps between the components. To avoid over-complication of the figure, only one of these maps is shown, that between the *left-road1* and *left-body1*.

Privits are transferred from the bottom series to the top: *decompose* and *move*.

Following is the control structure for our visual analogical transfer theory. We will describe the transfer of the first privit as a running example. The process in the abstract can be seen in Figure 1.

1. **Identify the first simages of the target and analog problems.**
2. **Identify the privits and associated arguments in the current simage of the source analog.** This step finds out how the source problem gets from the current simage to the next simage. In our example, the privit is *decompose*, with “four” as the *number-of-resultants* argument (not shown).
3. **Identify the objects of the privits.** The object of the privit is what object the privit acts on. For the *decompose* privit is the *soldier-path1* (the thick arrow in the bottom left simage.)
4. **Identify the corresponding objects in the target analog.** The *ray1* (the thick arrow in the top left simage) is the corresponding component of the source analog's *soldier-path1*, as specified by the analogical map between the simages (not shown). A single object can be mapped to any number of other objects. If the object in question is mapped to more than one other object in the target, then the privit is applied to all of them in the next step. If the privit *arguments* are components of the

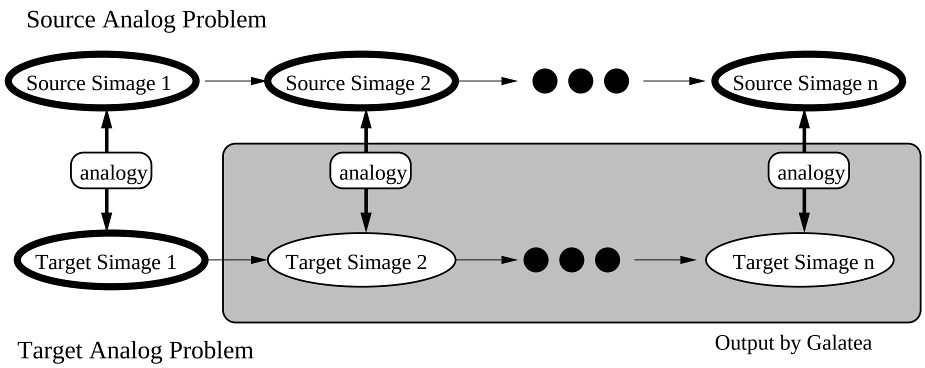

Figure 1: The things outside the shaded box are given to Galatea: a complete source problem an incomplete target problem, and the analogy between them. Galatea completes the analogical transfer and stores the new simage sequence for the target problem.

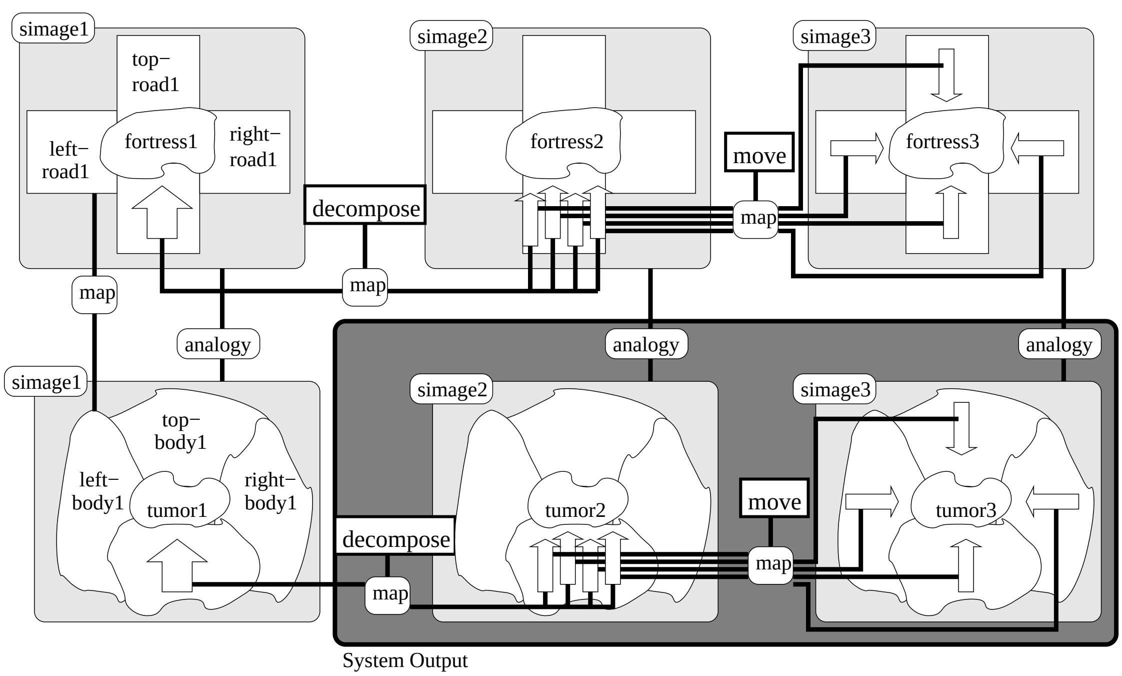

Figure 2: The fortress/tumor problem representation

source simage, then their analogs are found as well. Else the arguments are transferred literally.

5. **Apply the privit with the arguments to the target analog component.** A new simage is generated for the target problem (top middle) to record the effects of the privit. The *decompose* privit is applied to the *ray1*, with the argument "four." The result can be seen in the top middle simage in Figure 2. The new rays are created for this simage.

6. **Map the original objects to the new objects in the target problem.** A transform-connection and mapping are created between the target problem simage and the new simage (not shown). Maps are created between the corresponding objects. In this example it would mean a map between *ray1* in the first top simage and the four rays in the second top simage. The privit is associated with the map, as shown in the Figure, so the target problem itself can be used as a possible source analog in the future.

7. **Map the new objects of the target problem to the corresponding objects in the source problem.** In this case the rays of the second target simage are mapped to soldier paths in the second source simage. This step is necessary for the later iterations (i.e. going on to another transformation and simage). Otherwise the system would have no way of knowing which parts of the target simage the later privits would operate on.

8. **Check to see if goal conditions are satisfied.** If they are, exit, and the problem is solved. If not, and there are further simages in the source series, set the current simage equal to the next simage and go to step 1. If there are no further simages, then exit and fail.

3 System: Galatea

Our hypothesis was that a visual representation language would be sufficient to describe domains such that analogical problem solving could take place. To test this hypothesis, we implemented the above ideas in a program called Galatea, and applied it to Duncker's fortress/tumor analogy.

Galatea's knowledge representation architecture consists of two kinds of propositions: 1. A statement of existence of a concept or relation and 2. The connection of two concepts or propositions with a relation.

Galatea takes as input a solved source problem, an unsolved target problem (both represented visually), an analogical mappings between the simages, and criteria for an adequate problem solution. When instructed to solve the target using the source, it analogically transfers the solution procedure. As can be seen in Figure 2, it outputs a series of simages for the target problem, and checks to see if the solution transferred indeed solves the problem constraints. The following section describes our results.

Duncker's Fortress/Tumor Problem

Table 3 shows some of the privels and their attribute values for the first fortress problem simage.

Table 3: Privels from Fortress Problem Simage 1

Visual Object	attributes	value
Fortress	looks-like: location:	generic-visual-element center
Bottom-road	looks-like: start-point: end-point:	line bottom fortress
Right-road	looks-like: start-point: end-point:	line right fortress
Left-road	looks-like: start-point: end-point:	line left fortress
Top-road	looks-like: start-point: end-point:	line top fortress
Soldier-path	looks-like: location: thickness:	line bottom-road thick

We represented the fortress story with three simages (see Figure 2.) The first was a representation of the original fortress problem. It had four roads, represented as thick lines, radiating out from the fortress, which was a *generic-visual-element* in the center. We represented the original soldier path as a thick line on the bottom road. This simage was connected to the second with a *decompose* privit, where the arguments were *soldier-path1* for the *object* and "four" for the *number-of-resultants*. The second simage shows the *soldier-path1* decomposed into four thin lines, all still on the bottom road. The lines are thinner to represent smaller groups. This is connected to the final simage with the *move* privit, which is applied to three of the new soldier paths. They are sent to the different roads. The final simage in the fortress problem shows all four soldier paths, each on a different road.

We represented the start state of the tumor problem as a single simage. The tumor itself is represented as a *generic-visual-element*. The *ray* of radiation is a thick *line* that passes through the bottom body part.

Galatea transfers the first transformation (*decompose*) from the source analog (the solved fortress problem) to the target (the tumor problem). It knows which part of the tumor problem to apply this privit to from the given analogical mapping between the first simages of the fortress and tumor problems. Galatea generates a second simage with the *line* representing the ray decomposed into four thinner lines. In the next iteration Galatea successfully transfers the second transformation, moving each of the rays to the different roads.

Galatea can solve analogical transfer problems using only visual knowledge, as we have shown with the fortress/radiation example. Though this work is still in progress, we conjecture that this theory, when Privlan is more fleshed out, will apply to all problems whose solution constraints involve visually perceivable states of the world. Another sense of this is: if you can make a diagram of it, our theory applies to it.

3.1 Causal Knowledge

Though the solution procedure was transferred in both of these cases, the system still had no way of knowing if the transferred solution was *adequate* for the new problem. In the tumor problem, in order for the agent to determine if the tumor was destroyed and the patient was still alive, it needed some causal knowledge. By causal we mean knowledge of how things in a system change as they interact. Pre- and post-conditions are a straightforward way to represent this, but it is difficult to imagine what "visual" pre- and post-conditions might look like. Visual representations alone cannot enable evaluation of the solution.

Galatea represents causal knowledge with production rules, implemented in ACT-R [Anderson and Libiere, 1998]. We have no theoretical commitment to production rules or ACT-R. One production rule identifies a body part as dead if there is a thick line representing a ray going through it. Another rule identifies the tumor being killed if enough radiation is hitting it. If the tumor is dead and the body is alive, a final production fires that identifies the problem as being solved.

When the tumor problem is first encountered (when it only consists of a single simage), Galatea is unable to infer through the productions that the problem is solved in the initial state. When the solution is transferred from the fortress, the rules confirm that the problem has been solved.

4 Discussion

In our earlier work, we have developed a theory of Model-Based Analogy based on Structure-Behavior-Function models of causal mechanisms and physical systems. The IDeAL system [Bhatta and Goel, 1997], for example, transfers generic teleological mechanisms from a source analog to a target problem to address novel design problems. The ToRQUE system [Griffith *et al.*, 2000] uses generic structural transformations to mutate a target problem or a source analog to construct analogies. Galatea builds on the above theory of model-based analogy in that in it too relies on the core idea of generic transformations. Thus, while the analogical process in Galatea is similar to that in IDeAL, the content of its generic transformations is visual as opposed to teleological or structural. ToRQUE's structural knowledge captures only a small subset of visual knowledge. In contrast Galatea has information about the location and appearance of objects in a particular simage: the fortress is not just connected to the road, it is in the center of the simage; the path is not just on the road, it is a thick line. These additional features enable the initial analogical mapping between simages without causal knowledge because the simages representing the two analogs are similar when described visually.

Like Galatea, LetterSpirit is a model of analogical transfer [McGraw and Hofstadter, 1993]. It takes a stylized seed letter as input and outputs an entire font that has the same style. It does this by determining what letter is presented, determining how the components are drawn, and then drawing the same components of other letters the same way. Like Galatea, the analogies between letters are already in the system: the vertical bar part of the letter "d" maps to the vertical bar in the letter "b," for example. A mapping is created for the input character. For example, the seed letter may be interpreted as an "f" with the cross-bar suppressed. When the system makes a lower-case "t," by analogy, it suppresses the crossbar.

It is not at all clear that LetterSpirit is applicable to other domains (such as the fortress/tumor problem) in part because there is little distinction between its theory and the implementation that works for letters. In contrast, one can see how Galatea might be applied to the font domain: The stylistic guidelines in LetterSpirit, such as "crossbar suppressed" are like the visual transformations in Galatea: it would be a transformation of removing an element from the image, where that element was the crossbar and the image was a prototype letter "f." Then the transformation could be applied to the other letters one by one. We conjecture that our theory has more generality than LetterSpirit.

Galatea does not generate the analogical mapping, but other systems, that create mappings with visual information, have shown that it can be done. The VAMP systems are analogical mappers as well [Thagard *et al.*, 1992]. VAMP.1 uses a hierarchically organized symbol/pixel representation. It superimposes two images, and reports which components have overlapping pixels. VAMP.2 represented images as agents with local knowledge. Mapping is done using ACME/ARCS [Holyoak and Thagard, 1997], a constraint satisfaction connectionist network. The radiation problem mapping was one of the examples to which VAMP.2 was applied.

The Structure Mapping Engine, or SME [Falkenhainer *et al.*, 1990] finds the best mapping of elements between two domains. But SME typically is applied to instances where the situations are represented as having causal and structural knowledge. SME has been applied to visual knowledge in a system called MAGI [Ferguson, 1994], which takes visual representations and uses SME to find examples of symmetry and repetition in a single image.

Like Galatea, MAGI and the VAMPs use visual knowledge. But unlike Galatea their focus is on the creation of the mapping rather than on transfer of a solution procedure. MAGI's and Galatea's theories are compatible: a MAGI-like system might be used to create the mappings that Galatea uses to transfer knowledge. The theory behind the VAMPs is incompatible because they use a different level of representation for the images.

Galatea has also been applied to the case study of James Clerk Maxwell's creation of his electro magnetic theory. According to Nersessian's Cognitive-Historical Analysis [Nersessian, 1995], Maxwell used analogical transfer to resolve a problem with his mental model of electro-magnetism. The transfer was mediated by a generic abstraction, and the abstraction was created, retrieved, and instantiated using visual representations and reasoning.

Galatea so far has been substantiated for only two examples: Duncker's fortress/tumor problem and Maxwell's case study. In the future, we will extend Galatea to cover many more problems, and expand it to use, in addition to simages, bitmap images, which we believe will be important for changing representations when symbol mismatches make analogical mapping difficult.

5 Conclusion

The first finding of our experiments with Galatea is that visual knowledge alone, with no explicit representation of causal knowledge, is sufficient for enabling analogical transfer. This validates the central hypothesis of our work. Galatea suggests a computational model of analogy based on dynamic visual knowledge that complements traditional models based on causal knowledge. Although Galatea does not address the issues of retrieval and mapping, put together with other work described in the previous section, we can now more confidently conjecture that visual knowledge alone can enable retrieval, mapping and transfer in analogy.

A second finding of our work on Galatea is that evaluation, in general, cannot be done using visual knowledge alone; it requires causal knowledge too. Thus visual knowledge enables only the steps that depend directly on the visual similarity between the target problem and the source analog, e.g., retrieval, mapping and transfer. It does not, however, fully support the evaluation step because it depends not on similarity but on the intrinsic causal and teleological structure of the target problem.

Galatea represents visual knowledge symbolically, in the form of symbolic images made of primitive visual elements and primitive visual transformations. The symbolic representation provides the standard benefits of discreteness, abstraction, ordering, and composition. Although sequences of lower-level *bitmap* representations also capture the notion of ordering, they, by themselves, neither capture abstractions that enable noticing visual similarity nor enable transformations on the images. This leads us to a third finding: Galatea provides additional evidence that symbolic representations of visual images are necessary for analogy.

6 Acknowledgements

This work has benefitted from many discussions with Nancy J. Nersessian at the Georgia Institute of Technology. This research has been inspired in part by the work of B. Chandrasekaran at the Ohio State University and Lawrence Barsalou at Emory University in addition to that of Nersessian. Goel also benefited from discussions with David Waltz at NECI during his 1998 sabbatical.

References

[Anderson and Libiere, 1998] John R. Anderson and Christian Libiere. *The Atomic Components of Thought*. Lawrence Erlbaum Associates, 1998.

[Bhatta and Goel, 1997] Sambasiva R. Bhatta and Ashok K. Goel. A functional theory of design patterns. In *Proceedings of Fifteenth International Joint Conference on Artificial Intelligence*, pages 294–300, Nagoya, Japan, August 1997.

[Duncker, 1926] K. Duncker (1926). A qualitative (experimental and theoretical) study of productive thinking (solving of comprehensible problems). *Journal of Genetic Psychology*, 33:642–708, 1926.

[Falkenhainer *et al.*, 1990] Brian Falkenhainer, Kenneth D. Forbus, and Dedre Gentner. The Structure mapping engine: algorithm and examples. *Artificial Intelligence*, 41:1-63, 1990.

[Ferguson, 1994] Ronald W. Ferguson. MAGI: Analogy-based encoding using regularity and symmetry. In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society* pages 283-288. Atlanta, Georgia: Lawrence Erlbaum Associates, 1994.

[Griffith *et al.*, 2000] Todd W. Griffith, Nancy J. Nersessian and Ashok K. Goel. Function-follows-form transformations in scientific problem solving. In *Proceedings of the Twenty-Second Annual Conference of the Cognitive Science Society*. Lawrence Erlbaum Associates, Mahwah, New Jersey, 2000.

[Holyoak and Thagard, 1997] Keith J. Holyoak and Paul Thagard. The analogical mind. *American Psychologist*, 52(1):35–44, 1997.

[Kosslyn, 1994] Kosslyn, S. M. *Image and Brain: The Resolution of the Imagery Debate.* MIT Press, Cambridge, MA, 1994.

[McGraw and Hofstadter, 1993] G. McGraw and Douglas R. Hofstadter. Perception and creation of alphabetic style. In *Artificial Intelligence and Creativity: Papers from the 1993 Spring Symposium*, AAAI Technical Report SS-93-01, AAAI Press, 1993.

[Nersessian, 1995] Nersessian, N. J. Opening the black box: Cognitive science and the history of science. In Thackray, A. (ed.) *Constructing Knowledge in the History of Science. Osiris* 10, 1995.

[Pedone *et al.*, 1999] R. Pedone, John E. Hummel, and Keith J. Holyoak. The use of diagrams in analogical problem solving. Manuscript draft, 1999.

[Thagard *et al.*, 1992] Paul Thagard, D. Gochfeld, and S. Hardy. Visual analogical mapping. In *Proceedings of the 14th Annual Conference of the Cognitive Science Society*, pages 522–527, Hillsdale, Erlbaum, 1992.