# Representation Issues in Visual Analogy[1]

**Jim Davies (jim@jimdavies.org)**
**Ashok K. Goel (goel@cc.gatech.edu)**
Artificial Intelligence Laboratory, College of Computing;
Georgia Institute of Technology; 801 Atlantic Drive
Atlanta, GA 30332-0280 USA

**Abstract**

Visual analogies are analogies based on visual similarity. *Galatea* is a computer program that addresses the transfer task in visual analogies in the context of problem solving. Each source case in Galatea contains a problem-solving procedure, represented as a series of knowledge states and transformations between them. Source cases and target problems are represented in a symbolic language whose primitives pertain only to spatial objects and relations, and operations on them. Given visual representations of a source case and a target problem, and a mapping between the first knowledge state in the source and the target, Galatea adapts and transfers the problem solving procedure in the source to the target. In this paper, we describe some representation issues that arose in developing Galatea and its answers to them.

## Introduction

Imagine that a cognitive agent is trying to figure out how to put a battery into a tape recorder, and has access to a source case in which film is put into a camera. One way the two situations are similar is that they visually resemble each other: the battery and the film canister are shaped like cylinders, and the tape recorder and the camera are shaped like rectangular prisms with cylindrical holes in them. This visual similarity is more relevant to the problem than, say, any functional similarity between the devices in the two situations. This is an example of visual analogy, i.e., an analogy based on visual similarity.

One issue in visual analogy is how might an agent use visual similarity between two situations to transfer the problem-solving procedure in a source case to the target problem? Note that this issue is more general than the hypothetical example mentioned above. For example, source cases in many design domains contain drawings, diagrams, animations, photographs, videos, etc., and instructions for assembling complex artifacts often are presented to people in a completely diagrammatic from. Thus, establishing transfer of problem solutions using visual knowledge is a fairly general task.

*Galatea* is a computer program that addresses the transfer task in visual analogies in the context of problem solving. The development of Galatea raised several representation issues such as the modality of the representation, and the levels of abstraction and aggregation in the representation. These issues are common to many content-based theories of analogical reasoning. In this paper, first we briefly describe Galatea and then discuss the representation issues that arose in developing it.

## Galatea: A Computer System

Galatea is an operational program written in LISP. It implements the transfer of problem-solving procedures between visual analogs. The problem-solving procedure contained in a source case is represented as a series of knowledge states and transformations between them. Each knowledge state is represented as a symbolic image or *s-image*. The reasoner takes as input a source case, an initial knowledge state in the target problem, and an analogical mapping between the s-image representing the first knowledge state in the source case and the initial knowledge state in the target problem. Galatea adapts and transfers the visual transformations from the source to the target, creating new target s-images along the way. Figure 1 illustrates Galatea's input and output for the Duncker (1926) fortress/tumor problem.[2]

*Covlan* (the *Cognitive Visual Language*) provides Galatea with an ontology of visual primitives and transformations. Covlan's ontology of *primitive vi-*

[1]Davies, J., & Goel, A. K. (2003). Representation issues in visual analogy. in R. Alterman and D. Kirsh (Eds.) *Proceedings of The 25th Annual Conference of the Cognitive Science Society.* ©2003 The Cognitive Science Society

[2]In the Duncker (1926) problem, subjects read a story about a general who must overthrow a dictator in a fortress. His army is poised to attack along one of many roads leading to the fortress when the general finds that the roads are mined such that large groups passing over will set them off. To solve the problem, the general breaks the army into smaller groups and they take different roads simultaneously and arrive together at the fortress. Participants are then given a tumor problem, in which a tumor must be destroyed with a ray of radiation, but the ray will destroy healthy tissue on the way in, killing the patient. The analogous solution is to have several weaker rays converging on the tumor.

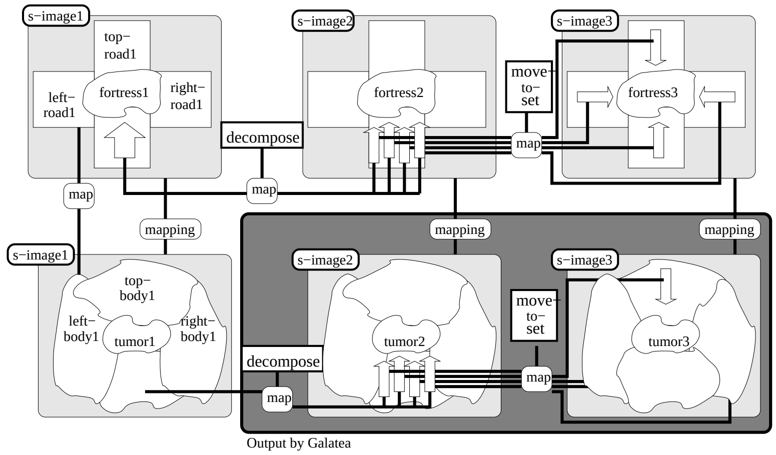


Figure 1: This Figure shows Galatea's input and output for the Duncker problem. The top series of s-images in the Figure shows the visual representation of the solved fortress problem. The bottom series shows the target tumor problem. The bottom left s-image is the initial state of the tumor problem. The shaded box shows the output of the system.

*sual elements* includes: *polygon, rectangle, triangle, ellipse, circle, arrow, line, point, spline,* and *text.* The elements are frame-like structures with slots that can hold values. For example, a *triangle* has a *location, size, height, width,* and *orientation.* Each transformation In Covlan is a operation (function) with arguments. Most transformations operate on some object, and many have additional arguments as well. These transformations implement normal graphics manipulations such as translation, rotation, scaling, and adding and removing visual elements.

Figure 2 illustrates a portion of final s-image in the tumor series generated by Galatea for the Duncker problem. The representation consists of a series of propositions, indicated in the Figure as labeled arrows connecting two elements. The objects in the s-image each have a location and are connected to a primitive visual element type with a *looks-like* relation. Each ray, represented as an arrow, also has a *thickness* — in this s-image, *thin.* Each arrow also has start and end points, also with locations (not shown in the figure). Not shown in figure are the maps that connect the elements of this s-image to the previous s-image, as well as the maps to the analogous source s-image.

## Algorithm

1. **Identify the first s-images of the target and source cases.**

2. **Identify the transformations and associated arguments in the current s-image of the source case.** This step finds out how the source case gets from the current s-image to the next s-image. In the Duncker example, the transformation is *decompose*, with *four* as the *number-of-resultants* argument (not shown).

3. **Identify the objects of the transformations.** The object of the transformation is what object the transformation acts upon. For the *decompose* transformation is the *soldier-path1* (the thick arrow in the top left s-image in Figure 1.)

4. **Identify the corresponding objects in the target problem.** The *ray1* (the thick arrow in the bottom left s-image) is the corresponding component of the source case's *soldier-path1*, as specified by the correspondences between the s-images (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 transformation is applied to all of them in the next step.

5. **Apply the transformation with the arguments to the target problem component.** A new s-image is generated for the target problem (bottom middle) to record the effects of the transformation. The *decompose* transformation is applied to the *ray1*, with the argument *four.* The result can be seen in the bottom middle s-image in Figure 1. The new rays are created for this s-image. Adaptation of the arguments can happen in three ways, as described above: If the argument is an object of the source s-image, then its analog is found. If the argument is a function, then the function is run (note that the function itself may have arguments which follow the same adaptation rules as transformation arguments). Else the arguments are transferred literally.

6. **Map the original objects to the new objects in the target case.** A transform-connection and mapping are created between the target problem

s-image and the new s-image (not shown). Maps are created between the corresponding objects. In this example it would mean a map between *ray1* in the left bottom s-image and the four rays in the second bottom s-image. This system does not solve the mapping problem, but a mapping from the correspondences of the first s-image enable the mappings for the subsequent s-images to be automatically generated.

7. **Map the new objects of the target case to the corresponding objects in the source case.** Here the rays of the second target s-image are mapped to soldier paths in the second source s-image. This step is necessary for the later iterations (i.e. going on to another transformation and s-image). Otherwise the reasoner would have no way of knowing on which parts of the target s-image the later transformations would operate.

8. **Check to see if goal conditions are satisfied.** If they are, exit, and the problem is solved. If not, and there are further s-images in the source case, set the current s-image equal to the next s-image and go to step 1. If there are no further s-images, then exit and fail. Goal conditions are represented non-visually (Davies & Goel, 2001).

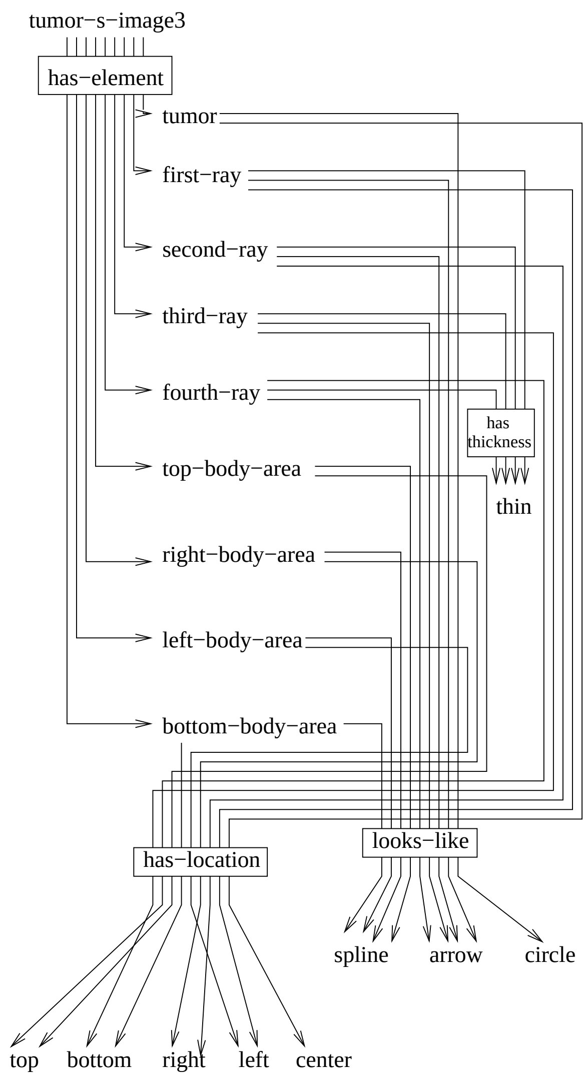


Figure 2: A portion of the third s-image in the tumor series.

Galatea presently works on three problems: the Duncker problem, a case study of scientific analogical reasoning by James Clerk Maxwell (Nersessian, 1984, Davies et al., 2003), and a cake/pizza problem in which a single pizza must be distributed among several people. We are currently extending the representational capacity and inferential capability of Galatea to cover a range of human subjects solving problems similar in complexity to the Duncker problem.

## Representation Issues in Galatea

The development of Galatea raised several representation issues including the modality of representation, the level of abstraction of the representation, the level of aggregation of the representation, and arguments to the visual transformations. Due to limitations of space, in this paper, we will briefly address only the first three issues.

### Issue 1: Modality of Representation

We view knowledge of different kinds as lying on a spectrum, at one end of which is raw sensory data (e.g. visual knowledge such as pixels in a photograph) and at the other extreme is highly interpreted and abstracted knowledge (such as causal or teleological knowledge). Visual knowledge higher up in the spectrum may be represented using symbolic structures. Galatea is a content-based theory of analogical problem solving, and thus makes a commitment about the *modality* of the representation. Some earlier content-based theories of analogical reasoning use causal and functional abstractions (e.g., Winston, 1980 for a functional representational account of the Duncker problem), while other content-based theories (e.g. Evans, 1968) use visual abstractions. The question for Galatea then is where on this spectrum should its representation fall?

### Issue 2: Representation's Level of Abstraction

There are several aspects to this issue. One aspect of this issue is whether to use symbolic, descriptive representations or pictorial, depictive representations (Kosslyn, 1994). Another aspect of the abstraction issue pertains to the level of abstraction in the symbolic representations. Marching armies and rays of radiation have their differences and similarities, and whether or not they can be used in analogical reasoning depends on the reasoner having representations of them at the same level of abstraction. Functionally, *soldier-path* is a different symbol from *ray*, and but they can both be conceptualized as *destructive-forces* (Holyoak & Thagard, 1989). Two similar ideas represented at very different levels of abstraction can lead to failures in retrieval, mapping, and transfer.

### Issue 3: Representation's Level of Aggregation

Yet another aspect of the abstraction issue is the level of aggregation at which things are represented. Suppose the reasoner's representation of the fortress story involved twelve roads, but in contrast the reasoner imagines a patient as having four areas of the body that rays might pass through. The different numbers of roads and body areas in the two analogs can cause problems with retrieval and mapping because there are some roads in the fortress story with no unmapped body areas in the tumor problem. An earlier version of Galatea (Davies & Goel, 2001) handled this by transferring two transformations: *decompose*, which broke thick lines (representing the army and ray) into four thinner lines (representing smaller armies and weaker rays), and *move-to-location*, which moved each individual thinner line to specific locations–a different road or area of the body. The program was brittle because it required there be the same number of roads as body areas.

## Galatea's Answers to the Issues

Galatea's answers to these issues evolve from our earlier work on analogical reasoning including the KRITIK (Goel, 1991; Goel, et al., 1997), IDeAL (Bhatta & Goel 1997a; Bhatta & Goel 1997b), and ToRQUE (Griffith et al., 2000; Griffith et al., 1996) systems. A central and persistent theme in our work has been that an analogical reasoner uses representations that reduce the complexity of retrieval, mapping, transfer, evaluation and storage. The appropriate representations may either be programmed into the reasoner (as in ToRQUE) or dynamically generated (as in IDeAL).

### Answer 1: Modality of Representation

Galatea uses visual abstractions to represent analogs. Thus, Galatea reasons about visual abstractions such as lines and arrows, and not armies and rays of radiation with all the semantic meanings typically associated with them. There are psychological and historically documented reasons to think visual representations are important for problem solving (e.g. Casakin & Goldschmidt, 1999; Nersessian, 1984; Pedone et al., 2001). We posit that visual abstractions provide a level of abstraction at which two otherwise dissimilar domains may be more alike (Davies et al., in press 2003). Galatea finds the *ray* and *soldier-path* similar not because they are both, for example, destructive forces, but because they are both *arrows.*

### Answer 2: Representation's Level of Abstraction

Galatea's symbolic, descriptive representations provide the standard benefits of directness, ordering, structure and composition.

Further Galatea's design opts for higher-level visual abstractions when possible. It could have, for example, used a complex hypothetical shape, say s1, to accurately describe the shape of a fortress, and a different different shape, say s14, to accurately represent the shape of a tumor. In that case, the tumor query might retrieve only other similar-looking tumors, ignoring even different-looking tumors, let alone fortresses. That is, a more detailed and more accurate visual representation would make analogical remindings, mappings and transfer harder. Thus, the current version of Galatea uses a higher-level representation for the tumor and the fortress. In a future version of Galatea, we plan to represent objects, such as fortresses and tumors, at multiple levels of abstractions and link the abstractions levels through shape hierarchies.

### Answer 3: Representation's Level of Aggregation

Recall that in Duncker problem, after the *decompose* transformation generates a number of smaller armies (represented in Covlan as thinner *lines*), they must be dispersed to the various roads, in various locations in the image. As mentioned above, in a previous version of Galatea each army line was *moved-to-location* individually to each road line. This solution was brittle because the number of roads the armies moved to needed to match exactly the number of body areas the weaker rays moved to in the target.

To overcome this limitation, the current version of Galatea uses the notion of *sets* to group armies, roads, rays, and body parts into their own different sets. The system now can adapt the solution during transfer to accommodate differing numbers of any of these elements. Rather than using the *move-to-location* transformation on each army, Galatea applies a new transformation *move-to-set* to the *set* of armies. The argument to this function is the set of roads. The *move-to-set* function takes one set and distributes its elements around the locations of another set. This robustness allows the transfer to occur even if the numbers of armies, rays, roads, and body parts are *all* different.

## Discussion

Some theories of analogical reasoning are feature-based while others are relation-based. For example, in some (but not all) theories of case-based reasoning, remindings are based on the features in the description of the target problem and adaptation too is limited to tweaking the features of the retrieved source case. In contrast, in relation-based theories of analogical reasoning, such as SME (Falkenhainer et al., 1990) and LISA (Gick & Holyoak, 1996), the focus and emphasis is on transfer of complex relations from the source to the target. Galatea too focuses and emphasizes the transfer of complex relations. In particular, it addresses the problem of transferring problem-solving procedures which contain an ordered series of operations.

Some relation-based theories of analogical reasoning are structure-based while others are content-based. SME, for example, provides an uniform structure-based mechanism for analogical reasoning that is intended to work independent of any specific content account. Content-based theories, such as (Winston, 1980), ANALOGY (Evans, 1968), and LetterSpirit (McGraw & Hofstadter, 1993), focus and emphasize the content of the representations, and the mechanisms of analogical reasoning are content-dependent. Galatea too is a content-based theory of analogical reasoning.

Among the content-based theories of analogical reasoning, most accounts make use of causal and functional knowledge, for example, (Winston, 1980) and our own earlier work on KRITIK, IDeAL, and ToRQUE. The IDeAL system, for example, uses structure-behavior-function (SBF) models for supporting analogical remindings, mappings and transfer in the context of conceptual design. The ToRQUE system uses SBF models for analogical remindings, transfer and evaluation in the context of scientific problem solving. In contrast, Galatea's mechanism is driven by a content account of visual abstractions and aggregations.

Like Galatea, ANALOGY and LetterSpirit are content-based theories of analogical transfer using visual representations. ANALOGY is an early computer program that performed visual analogies. It solved multiple choice visual analogy problems of the kind found on intelligence tests (e.g. A:B::C:?). It does this by describing how to turn A into B, and then testing into which choice might C turn into in a similar manner. LetterSpirit 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. 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.

Neither ANALOGY nor LetterSpirit transfers problem-solving procedures (ordered series of operations) as Galatea does. In contrast, one can see how Galatea might be applied to, say, LetterSpirit's domain: The stylistic guidelines in LetterSpirit, such as "crossbar suppressed" are like the visual transformations in our theory: crossbar suppressed would be a transformation of removing an element from a knowledge state, where that state was a prototype letter *f* and the element was the crossbar. This transformation then could be applied to the other letters one by one.

Other systems have approached different aspects of the Duncker analogy. Diva, for example, (Croft & Thagard, 2002) does analogical mapping, using ACME as the infrastructure.

The visual primitives that describe a knowledge state in Galatea are similar to that of GeoRep (Ferguson & Forbus, 2000). Galatea, however uses them for a task quite different from that of GeoRep: GeoRep extracts and abstracts visual relations in line drawings; in contrast, Galatea transfers a problem-solving procedure from a source case to a target problem. To do so, in addition to the visual primitives for describing a knowledge state, Galatea uses primitive transformations that act on knowledge states. Galatea's notion of groups too is similar to that GeoRep. GeoRep dynamically generates groupings for abstracting visual relations from line drawings while Galatea uses groups to enable analogical transfer.

## Conclusion

Three core issues in content-based accounts of analogical problem solving are the modality of representation and the representation's levels of abstraction and aggregation. Most content accounts of analogical problem solving are at the level of causal and functional knowledge, i.e., reminding, mapping, adaptation and transfer are based on causal and functional abstractions. In contrast, Galatea's content account is at the level of visual abstractions. Galatea shows that, in some situations, visual knowledge alone is sufficient for the transfer task.

Galatea represents visual abstractions symbolically. The system's design opts for higher-level visual abstractions whenever possible. In future versions of Galatea, we plan to multiple levels of visual abstractions. The current version of Galatea uses multiple levels of aggregation. In particular, it uses the notion of sets and set membership to group visual primitives, which facilitates analogical transfer.

## Acknowledgments

This work has benefited from many discussions with Nancy J. Nersessian.

## References

Bhatta, S. and Goel, A. K. (1997a) A functional theory of design patterns. In the *Proceedings of IJCAI-97.* 294–300.

Bhatta, S. and Goel, A. K. (1997b) Learning Generic Mechanisms for Innovative Design Adaptation. *Journal of Learning Sciences,* 6(4):367-394.

Casakin, H, & Goldschmidt, G. (1999). Expertise and the use of visual analogy: Implications for design education. *Design Studies*, 20:153–175.

Croft, D. & Thagard, P.(2002) Dynamic imagery: A computational model of motion and visual analogy. In L. Magnani (Ed.), *Model-based reasoning: Scientific discovery, technological innovation, values.* New York: Kluwer/Plenum.

Davies, J., & Goel, A. K. (2001). Visual analogy in problem solving. *Proceedings of the International Joint Conference on Artificial Intelligence 2001.* pp 377-382. Morgan Kaufmann publishers.

Davies, J., Goel, A. K. & Nersessian, N. J. (in press 2003). Visual Re-Representation in Creative Analogies. *The Third Workshop on Creative Systems. International Joint Conference on Artificial Intelligence.* Acapulco, Mexico.

Davies, J., Nersessian, N. J., & Goel, A. K. (2003). Visual models in analogical problem solving. To appear in *Foundations of Science 2003, special issue on Model-Based Reasoning: Visual, Analogical, Simulative.* Magnani, L. & Nersessian, N. J., Eds.

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

Evans, T. G. (1968). A heuristic program to solve geometric analogy problems. In *Semantic Information Processing* edited by Marvin Minsky MIT Press, Cambridge, MA.

Falkenhainer, B., K. D. Forbus, & D. Gentner (1990). The Structure mapping engine: algorithm and examples. *Artificial Intelligence* (41) 1–63.

Ferguson, R. W. & Forbus, K. D. (2000) GeoRep: A flexible tool for spatial representation of line drawings, *Proceedings of the 18th National Conference on Artificial Intelligence.* Austin, Texas: AAAI Press

Gick, M. L. & K. J. Holyoak. (1996). LISA: A computational model of analogical inference and schema induction. In *Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society*, 352–357. Atlanta, GA: Lawrence Erlbaum Associates.

Goel, A. K. (1991). A Model-Based Approach to Case Adaptation. In *Proc. Thirteenth Annual Conference of the Cognitive Science Society*, Chicago, 143–148, Hillsdale, NJ: Lawrence Erlbaum.

Goel, A. K., Bhatta, S. R., & Stroulia, E. (1997). KRITIK: An early case-based design system. In: *Issues and Applications of Case-Based Design.* Maher, M. L. and Perl, P., eds., Erlbaum, Hillsdale, NJ., 87–132.

Griffith, T. W., Nersessian, N. J. & Goel, A. K. (1996). The role of generic models in conceptual change. In *Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society*, Lawrence Erlbaum, Mahwah, NJ.

Griffith, T. W., Nersessian, N. J., & Goel, A. K. (2000). Function-follows-form transformations in scientific problem solving. *Proceedings of the Twenty-Second Annual Conference of the Cognitive Science Society.* Lawrence Erlbaum, Mahwah, NJ.

Holyoak, K. J., & Thagard, P. (1989). A computational model of analogical problem solving. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning.* Cambridge: Cambridge University Press. 242–266.

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

McGraw, G. & Hofstadter, D. R. (1993) 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.

Nersessian, N. J. (1984) *Faraday to Einstein: Constructing Meaning in Scientific Theories.* Kluwer, Dordrecht, pp. 68-93.

Pedone, R., Hummel, J. E., & Holyoak, K. J. (2001). The use of diagrams in analogical problem solving. *Memory & Cognition*, 29, 214–221.

Winston, P. H. (1980). Learning and reasoning by analogy. *Communications of the ACM.* 23:12.