Qualitative Spatial Reasoning for Visual Grouping in Sketches

Kenneth D. Forbus Emmett Tomai Jeffrey Usher
Qualitative Reasoning Group
Northwestern University
1890 Maple Avenue,
Evanston, IL, 60201 USA
847-491-7699
forbus@northwestern.edu

Abstract

We believe that qualitative spatial reasoning provides a bridge between perception and cognition, by using visual computations to construct structural descriptions that have functional significance. We provide evidence for this hypothesis by describing how qualitative spatial reasoning can be used to model aspects of visual structure in sketches. We begin by outlining the nuSketch spatial reasoning architecture, including the representation of glyphs and sketches and the use of qualitative topology and Voronoi diagrams to construct spatial representations. We then describe our use of *spatial analogies* as a means for exploring the structure of visual representations. Three concepts of visual structure in sketches are introduced: *connected glyph groups, contained glyph groups*, and *positional relations*. We show that by using visual reasoning techniques to compute these qualitative descriptions, spatial analogies involving sketches are significantly improved.

1. Introduction

One of the mysteries of human cognition is how we make sense of the world around us. We have powerful visual systems, and it appears that part of their job is to compute descriptions of visual structure (cf. [22,23,11]) which can be used for recognition and understanding. We have argued previously that qualitative spatial reasoning plays an important role in medium and high-level visual processing [12]. Qualitative spatial representations provide a bridge between vision and cognition, since they seem to be computed via visual processes, but taking functional constraints into account. We have been exploring this idea by research on sketching. Understanding sketches is a useful approach to understanding visual structure because starting with digital ink lets us focus on processes of perceptual organization and ignore image processing issues. This paper describes some techniques we have developed for imposing human-like visual structure on sketches. We show that these techniques enable our software to better model human similarity judgments concerning sketches.

We start by reviewing our approach to sketching and the *sketching Knowledge Entry Associate* (sKEA) [15], an open-domain sketching system used in these experiments. Next we provide an overview of the spatial representations of sketches and glyphs and the processing architecture that handles spatial computations. Then we describe the computation of spatial relationships, including qualitative topology and Voronoi diagrams. Three kinds of visual structure, based on qualitative spatial representations, are introduced: *connected glyph groups*, *contained glyph groups*, and *positional relations networks*. We demonstrate that introducing these visual structures can improve analogies involving sketches. Finally, we discuss plans for future work.

2. Overview of nuSketch and sKEA

Sketching is a form of multimodal interaction, where participants use a combination of interactive drawing and language to provide high-bandwidth communication. Sketching is especially effective in tasks that involve space, e.g., physical structures or maps. While today's software is far being as fluent as sketching with a person, research on multimodal interfaces has produced interfaces that are significantly more natural than standard mice/menu systems (cf. [2]).

sKEA is designed to enable knowledge entry via sketching. Unlike most sketching systems, which are limited to a narrowly constrained domain, sKEA is open-ended: Any concept in its large knowledge base can be included in a sketch. Specifically, we use a subset of Cycorp's Cyc knowledge base contents[1], with extensions developed by our group for qualitative and analogical reasoning.

[1] We use our own KB and reasoning system instead of Cyc that is optimized for our needs. The subset of Cyc we use contains tens of thousands of concepts.

The typical approach in multimodal interfaces is (a) to provide a more natural interface to a legacy software system and (b) to focus on recognition [1,2]. While this approach has led to useful systems, it has some serious limitations. First, today's statistical recognizers are not very good (indeed, much of the multimodal literature focuses on using multiple modalities to overcome the limitations in individual modalities). Second, speech recognition requires that the vocabulary and grammar can be fixed in advance, and smaller vocabularies and grammars lead to more accurate recognition. This can be reasonable for sketching systems designed to operate in a tightly constrained domain, but for sKEA, which is designed to be general-purpose, such *a priori* restrictions are not possible. Third, even if recognition improves to human-level or beyond, there is still the problem of providing software with a visual and conceptual understanding of what is being sketched. Such knowledge is crucial for creating knowledge capture and performance-support systems.

Our approach in the *nuSketch architecture* [12] is quite different and complements traditional multimodal research. We avoid recognition issues by using clever interface design. We focus instead on providing richer visual and conceptual understanding of what is sketched. In addition to sKEA, we have created a second system based on this architecture, nuSketch Battlespace (nSB) [16], specialized for military reasoning. While nSB shares a common code base for spatial reasoning, we focus in this paper on sKEA for brevity.

sKEA's interface provides ways to enable users to specify conceptual information about the entities being sketched. (The interface techniques that enable us to avoid recognition are described in [16].) sKEA uses the knowledge base to draw additional inferences about the conceptual relationships depicted in the sketch (e.g., if a nucleus is drawn inside the body of a cell, that suggests that the nucleus could be part of the cell). Complex ideas, such as sequences or alternate points of view, can be conveyed using *subsketches* that are combined on the *metalayer* to form "the whole story". sKEA is still a research system, although we have carried out internal experiments where graduate students from other groups were able to use it successfully.

3. Representing glyphs and sketches

This section describes the underlying ontology of sketches that we use. The basic unit in a sketch is a *glyph*. Every glyph has *ink* and its *content*. The ink consists of one or more polylines, representing what the user drew when specifying that glyph. (Each polyline includes width and color information in addition to its points.) The content is a conceptual entity, the kind of thing that the glyph is representing. For example, if a user drew a ball, there would be an entity created to represent the glyph itself and an entity to represent the ball. While each subsketch depicting the ball would have a distinct glyph, the contents of those glyphs would all be the same entity.

While some basic spatial properties of glyphs are computed (described below), we do not perform any detailed shape reasoning on the ink comprising a glyph, nor do we attempt to visually decompose it. We call this *blob semantics* because it focuses on spatial relationships between glyphs rather than detailed reasoning about the visual structure of glyphs themselves. While inappropriate for recognition based on detailed visual similarity of specific features, it is an excellent approximation for many kinds of spatial reasoning, whenever the focus is on configural relationships between glyphs. Given the crude level of most people's artistic skills, they are unlikely to be extremely accurate at reproducing shapes.

A sketch consists of one or more *subsketches*. Subsketches represent a coherent aspect of what is being sketched, such as a state of a plan, or a more detailed depiction or distinct perspective on something. Logically, subsketches are Cyc-style microtheories, local descriptions that must be internally consistent. In sKEA, every subsketch has an associated *genre* and *viewpoint*. The genre specifies the overall type of the subsketch, and is one of `AbstractSketch`, `PhysicalSketch`, `GeospatialSketch`, or `DiscreteGraphSketch`. The *viewpoint* of a sketch describes the relationship between the visual frame of reference of the glyphs and the spatial frame of reference for the contents. Examples of viewpoint include `LookingFromTopView`, `LookingFromSideView`, `LookingFromBelowView`, and `LookingFromDirectionView`. Combinations of genre and view determine how visual relationships between the ink of glyphs translate into spatial relationships between their contents. For example, given a `PhysicalSketch` and `LookingFromSideView`, the same deictic user-centered vocabulary of spatial relationships (`above`, `below`, `leftOf`, `rightOf`) is assumed to be appropriate for both ink and contents. On the other hand, for a `GeospatialSketch` and `LookingFromTopView`, the vocabulary `eastOf`, `westOf`, `northOf`, and `southOf` is used instead. No inferences about spatial relations between contents are sanctioned by spatial relations between glyphs in the `AbstractSketch` and `DiscreteGraphSketch` genres.

Visually, the user sees either a single subsketch at a time, or the *metalayer*, a special view where each subsketch is viewed as a glyph. Relationships between subsketches can be entered by drawing labeled arrows between subsketch glyphs.

4. Spatial processing of glyphs

Spatial reasoning is carried out when a glyph is added, moved, or resized. sKEA has two visual processors, which are threaded to enable computation while the user is thinking or sketching. We describe each in turn, as a prelude to the detailed discussion of the spatial operations.

The *ink processor* is responsible for computing basic spatial properties of glyphs and responding to queries concerning spatial relationships. Whenever a glyph is added or changed, basic spatial properties are computed for it, including a bounding box, area, overall orientation and roundness. Relative size (comparing bounding box area to other glyphs in the subsketch) is also computed, classifying a glyph as either tiny, small, medium, large, or huge, using a logarithmic scale to determine size category boundaries. Qualitative topological relationships are automatically computed between the new glyph and other glyphs on its layer.

The *vector processor* is responsible for maintaining a set of Voronoi diagrams describing spatial relationships between types of entities, and for the polygon operations used in position-finding and path-finding. Any time a glyph is added or changed, once the ink processor has updated its properties the Voronoi diagram(s) it is associated with are updated appropriately. When spatial constraints involving position-finding or path-finding need solving, the vector processor carries out the construction of obstacle and cost diagrams, the polygon operations needed to combine them, and the quad tree representation used in path-finding. (Position-finding and path-finding will not be discussed further in this paper.)

Conclusions reached by these processors are added to the LTMS-based working memory of the reasoner for that sketch. The justifications include a "last changed" time-stamped assumption for each glyph involved. These assumptions are retracted whenever glyphs are moved, resized or deleted, which causes the conclusions that depend on the previous visual properties of the glyph to be automatically retracted.

5. Spatial relationships between glyphs

Spatial relationships are the threads from which configural information is woven. Therefore computing them appropriately is a crucial problem for qualitative reasoning about sketches. We discuss four kinds of spatial relationships in turn: Qualitative topological relationships, Voronoi relationships, positional relationships, and relationships based on local frames of reference.

5.1 Qualitative topological relationships

We use the RCC8 algebra [3] to provide a basic set of qualitative relationships between glyphs. RCC8 is appropriate because it captures basic distinctions such as whether or not two glyphs are disjoint (DC), touching (EC), or inside one another (TPP, NTPP). These distinctions are used in several ways. First, they are used in controlling when to compute other relationships: computing whether or not one entity is east of another is moot unless they are DC, for example. Second, they suggest conceptual interpretations of relationships between the contents of the glyphs that they relate. For instance, an EC relationship between two glyphs which represent physical objects suggests that their contents might be touching. Finally, domain-specific inference rules can use these relationships when needed, e.g., containment.

Much of the work on RCC8 and other qualitative topological algebras has focused on using transitivity for efficient inference. For sketches the use of such tables is unnecessary, because we can simply calculate for each pair of glyphs what RCC8 relationship holds between them, based on the visual properties of their ink. By default, we compute RCC8 relationships between a glyph and everything else on its subsketch when it is first added or changed.

5.2 Voronoi Relationships

Following [7], we use Voronoi diagrams to compute a variety of spatial relationships. Recall that, given a set of spatial entities (called *sites*, typically points), a Voronoi diagram consists of edges that are equidistant from a pair of points. The Delauney triangulation is the dual of the Voronoi, consisting of a set of arcs between sites that have an edge between them in the Voronoi diagram. As [7] describes, the Delauney triangulation provides a reasonable approximation to visual proximity, in that two sites are proximal exactly when there is an edge connecting them in the Delauney triangulation. Moreover, a number of approximations to spatial prepositions can be computed, including between and near. Again, these are approximations: It is known that, psychologically, spatial prepositions depend on functional and conceptual information as well as spatial information [5,10]. However, we have found them adequate for sketch maps.

Voronoi computations are defined in terms of sites being points, but glyphs have significant spatial extent. Consequently, we add a glyph to a Voronoi diagram by using sample points along the outer contour of the glyph's ink, each of which is treated as a site. These sites are marked with the glyph they derived from. While the Voronoi computations are done on the sampled sites, the results are expressed in terms of relationships between the glyphs. For example, two glyphs are `siteAdjacent` exactly when there exists a sample site on each glyph that is connected by an edge in the sample-level Delauney triangulation.

A key design feature in any system using Voronoi computations is what diagrams should be computed. Given sKEA's general-purpose nature, we currently use one Voronoi diagram per subsketch, which can be viewed as capturing the visual proximity between the ink of its glyphs. We suspect that in some cases multiple Voronoi diagrams will be needed for domain-specific reasoning (e.g., a Voronoi diagram consisting of only glyphs whose contents are physical entities, leaving out glyphs that represent purely conceptual entities), but we have not needed this level of complexity in sKEA yet.

5.3 Positional relationships

Positional relationships provide qualitative position and orientation information with respect to a global coordinate frame. Positional relationships between the ink of glyphs are expressed in a viewer-centered coordinate system of above/below, left/right in the plane of the sketch. As noted earlier, positional relationships for a sketch depicting physical entities seen from the side are expressed in the same relational system. Positional relationships between geospatial contents are expressed in terms of compass directions. For example, a playground can be south of a school and to the east of a street.

A key design choice is what positional relationships should be computed. It might seem at first that, like RCC8 relationships, it could be worth computing positional relationships between every pair of RCC8-DC glyphs. This turns out to be a terrible strategy. The computational load for computing them is not horrible, but the resulting network of relationships leads to inaccurate matches when doing spatial analogies. Essentially, computing every possible positional relationship reduces the distinguishability of different aspects of a sketch, since what makes the spatial positioning of a glyph unique is more a function of its local neighborhood than its global properties in the sketch. Thus the task of spatial analogies imposes a strong constraint on what should be computed in terms of spatial relationships. (The importance of this constraint for cognitive modeling is discussed further below.)

Computationally, positional relationships are used to provide concise summaries (if communicating a situation) and to provide a framework for describing the layout of a situation (for instance when computing spatial analogies). This framing function of positional relations suggests that they should respect the visual neighborhood structure of the sketch. Consequently, we use the Voronoi diagram for a subsketch to determine what positional relations to compute. The positional relation between a pair of glyphs is computed only when they are `siteAdjacent` in that subsketch's Voronoi diagram. (This is a necessary condition but not sufficient; the final condition relies on the grouping techniques described below so we postpone discussing it until then.) This has the desired effect of constructing a local network of positional relations.

How psychologically plausible is this design decision? There are, to be sure, cases where people construct on demand positional relations between entities that are quite distant. For example, in communicating the position of a location on a map, ignoring local neighborhood structure and describing it in terms of relationships to highly salient landmarks makes a lot of sense. Nevertheless, we suspect that the local scheme we have adopted reflects one of the default encoding techniques that people use in visual understanding.

Figure 1: Example of a sketch with glyph groups

6. Visual grouping

People naturally group visual entities using a variety of principles [22,23]. Our current focus on blob semantics places many of these techniques out of sKEA's scope. However, we can exploit the RCC8 relationships sKEA computes to detect at least two kinds of natural visual structure. The first, *connected glyph groups*, consist of a set of glyphs that are EC (i.e., edge connected) or PO (i.e., partially overlapping) with each other. We include PO because sketches can be inaccurate. An example of a connected glyph group is the head, ears, and body of the cat shown in Figure 1. The second, *contained glyph groups*, consist of glyphs that are directly inside another glyph (as indicated by TPPi and NTPPi – tangential proper part and non-tangential proper part inverse relationships). An example of a contained glyph group is the eyes, nose, and mouth within the head of the cat. Both rely on the Gestalt principle of contiguity: Connected glyph groups consist of a set of things that are touching, whereas contained glyph groups consist of a set of things bounded by another. sKEA maintains two intermediate graphs to compute these glyph groups. The *connection graph* for a subsketch consists of a graph whose nodes are glyphs and whose arcs are between pairs of glyphs that are currently EC or PO. The *containment graph* for a subsketch consists of a graph whose nodes are glyphs and whose arcs are between pairs of glyphs that are TPPi or NTPPi. The statement for each node of what links connect it to other elements of the graph is justified in terms of a closed-world assumption, in addition to the current ink assumptions for each of the glyphs involved. These closed-world assumptions are tested every time a glyph is added, moved, resized, or deleted from the subsketch, and the graphs are recomputed as necessary. Recomputing a subset of either graph causes the appropriate glyph group detection algorithm to be run on the changed subset.

Given the connection and containment graphs, finding glyph groups is straightforward. Every connected subset of the connection graph forms a connected glyph group. Contained glyph groups are found by the following algorithm:

1. For each glyph `G` such that `|arcs(ContainmentGraph(G))| > 2`,
2. Initialize *insiders* = `arcs(ContainmentGraph(G))`
3. For each $i \in$ *insiders*, let *internal* = `arcs(ContainmentGraph(i))`.
   3.1 Let *insiders* = *insiders* – *internal*
4. If |*insiders*| > 1, then create new contained glyph group C with `container(C,G)` and `insider(C,i)` for each *i* in *insiders*.

This algorithm ensures that only glyphs that are directly contained, as opposed to those nested inside yet some other container, are considered as part of a glyph group. This reflects our assumption that such perceptual organizations are applied recursively, at multiple scales.

Contained glyph groups are used to constrain the construction of positional relations. Recall that positional relations are only computed between pairs of glyphs that are DC and are `siteAdjacent` in the Voronoi diagram. One drawback to defining the Voronoi for glyphs in terms of the Voronoi determined by sample points along its contour is that there can be errors introduced by sampling, which produces "leaks" that can corrupt the neighborhood structure. People's inaccuracy when sketching can also cause errors in neighborhood structure, as determined by simple numerical calculations. For instance, a glyph G1 that is PO to a glyph G2 that contains a number of other glyphs can appear to be a neighbor to the glyphs inside G2. Visually, however, we would not consider them to be neighbors because G2 "blocks" them. We avoid both kinds of errors by an additional filtering constraint: If a glyph is in a contained glyph group, positional relations can only be computed with other members of the same glyph group. Thus we use the more robust qualitative topological computations to help avoid errors due to sampling and human inaccuracies.

Another use of glyph groups is to provide a context for relative size judgments. Just as the relative size of a glyph is characterized based on the other glyphs of the subsketch, additional relative size information is added based on the other glyphs in the group as the basis for comparison. Also, *articulation points* are computed for connected glyph groups, i.e., any glyph whose removal would completely disconnect the glyph. Visually such glyphs often represent a central piece that other things are connected to, e.g., the head of the cat which serves as an articulation point for the connected glyph group consisting of its ears and whiskers and torso.

7. Visual analogies

A key aspect of our approach is the use of human-like analogical processing for comparisons. Our goal is to ensure that, within the limitations of our representations, things which look alike to human users will look alike to the software. This *shared similarity constraint* enables the software's conclusions to be more trusted by users. We achieve a shared sense of similarity by using cognitive simulations of human analogical processing, over representations that approximate human visual representations. The cognitive simulation of analogical matching we use is the Structure-Mapping Engine (SME) [9], which is backed by considerable psychological evidence [17]. There is evidence that the structural alignment processes SME models are operating in human visual processing [11], which makes using it a reasonable choice.

The shared similarity constraint has proven to be a valuable constraint on representation and reasoning choices. For many pairs of sketches, the question "what goes with what" has clear and unambiguous answers for people viewing them. Suppose sKEA computes different correspondences. There are only three reasons this can occur (1) people are relying on visual properties that lie outside blob semantics, (2) sKEA's match process is operating differently than what people are doing, or (3) sKEA's representations differ in significant ways from the representations that people are using. We can rule out the first explanation by careful choice of sketches. The second explanation is ruled out by the existence of independent evidence for the use of structural alignment in high-level vision and SME's accuracy in modeling such structural alignment processing. Thus the explanation must lie in the representations that sKEA is computing being different in some significant way from what people are using. Thus the shared similarity constraint provides a kind of X-ray for exploring visual structure. We have used this technique to guide many of the representation and processing choices described in this paper. Next we will use it to demonstrate that our neighborhood method of computing positional relationships and our glyph grouping techniques help sKEA to perceive visual similarity in a more human-like way.

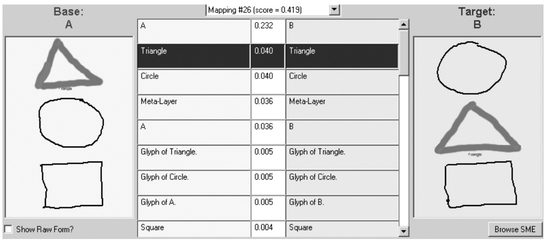

Figure 2: Similarity sans positional relations

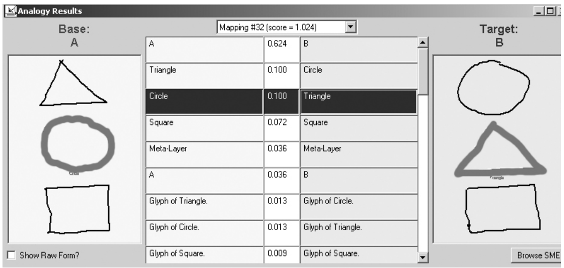

Figure 3: Similarity with positional relations

8. Experimental results

We have tested our techniques on a corpus of over a dozen pairs sketches to date. The contents of the sketches range from drawings of animals to maps to simple physical situations. Our hypotheses are that the neighborhood method of computing positional relations and our glyph grouping techniques are part of the high-level visual structure that people compute when looking at sketches, and consequently, they should improve spatial analogies when they are used. While our experiments are still in progress, we have at least some initial evidence to support these hypotheses. Here we summarize our results so far.

Positional relations can indeed help model psychological phenomena. In Figure 2, sKEA without positional relations behaves much as human

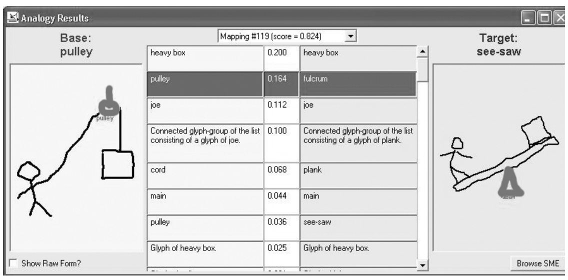

Figure 4: Multiple types of visual relationships can be needed for successful matching

subjects do when given very short times to make similarity judgments: They select object matches. Figure 3 shows that when positional relations are introduced, sKEA, like human subjects, prefers correspondences that are consistent with a larger relational system.

We have been running a variety of examples with different combinations of spatial relations to better understand what each contributes. For example, in Figure 4, grouping and positional relations are sufficient to cause the "end" objects to match (the person and the block). Adding articulation points enables the cord/plank comparison to be found, and adding in sizes for layers and connected groups provides a complete match.

In the case of glyph groups, the extra constraint they impose helps keep glyphs within the group matching to other glyphs within the group. Consider for example the cat head/person head comparison shown in Figure 5. Without glyph groups, the qualitative descriptions computed purely on the basis of individual blobs cause a variety of unintuitive matches. With glyph groups, most of the parts of the heads correspond as one would expect, as Figure 6 illustrates.

We have also found that these techniques can sometimes interact in negative ways. For example, using positional relations with some sketch pairs can lead to mismatches, as Figure 7 illus-

Figure 5: A poor analogy without grouping

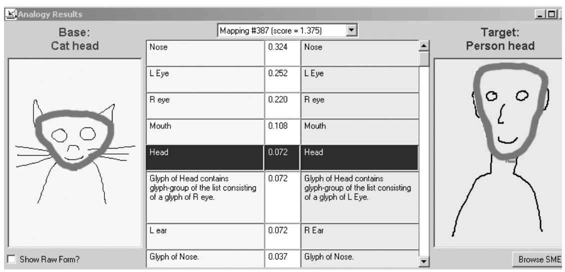

Figure 6: Grouping improves visual analogy

trates. Such negative competitions suggest that we have to either introduce yet more visual structure, or institute more fine-grained control over what gets computed when. For example, one technique that seems appropriate both from a visual psychology perspective and from improving matches is to treat glyph groups as new individuals, over which properties such as orientation and positional relations are computed.

9. Other Related work

Qualitative spatial reasoning has often focused on mechanical systems (cf. [14,25]), but some have

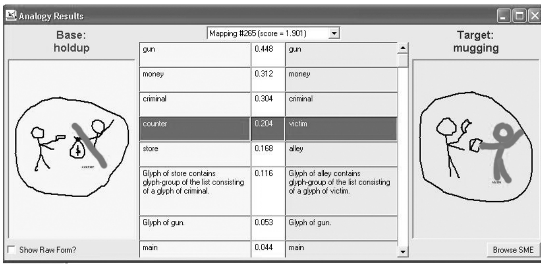

Figure 7: Positional relations can lead to mismatches

focused on navigation and locations (cf. [20]). Sketching research (e.g., [1,2,19]) tends to focus on tightly constrained domains, in order to keep recognition tractable. Several researchers have explored visual analogies, but typically have used *ad hoc* special-purpose matching algorithms, rather than a general-purpose model of analogical matching (cf. [6]). The hypothesis that qualitative spatial reasoning is involved in visual perception is also being explored by Cohn's group (cf. [4]), whose focus is on extracting qualitative descriptions of dynamic behaviors from camera data.

10. Discussion and Future work

We have argued that qualitative spatial representations serve as a bridge between perception and cognition. As evidence for this claim, we have shown that adding positional relationships computed on local (in the Voronoi sense) neighborhoods and two grouping techniques for glyphs (connected and contained) often improve the similarity of pairs of sketches, bringing them more in line with human judgments of the same sketches.

While these results are encouraging, much research remains. First, even under the simplifying assumption of blob semantics, we have not exhausted the perceptual organizations that people appear to compute. For example, some kinds of visual structure such as symmetry impose new frames of reference which are used to compute additional relationships. Moreover, it appears likely that some visual structuring is imposed based on the content, not just the ink of a glyph – consider for example entities with clear orientations, e.g., houses. We plan to explore a broader range of sketches to identify such possibilities, and follow-up experiments to hone our representations to capture them. Second, we plan on experimenting with sketch retrieval (cf. [19]), both to explore the nature of human encoding of sketches into long-term memory and to enable sKEA to have a shared history of sketches with its users. We plan to use our MAC/FAC model of similarity-based reminding [13] for this. Finally, we will ultimately need to move beyond the assumption of blob semantics, to tackle finer-grained shape descriptions and automatically decompose glyphs accordingly. This will take careful study of the vision science literature to constrain the process as tightly as possible (cf. [11]).

Acknowledgements

This research was supported by the DARPA Rapid Knowledge Formation program and by an National Science Foundation ROLE grant.

References

1. Alvarado, Christine and Davis, Randall (2001). Resolving ambiguities to create a natural sketch based interface. *Proceedings of IJCAI-2001*, August 2001.
2. Cohen, P. R., Johnston, M., McGee, D., Oviatt, S., Pittman, J., Smith, I., Chen, L., and Clow, J. (1997). QuickSet: Multimodal interaction for distributed applications, Proceedings of the Fifth Annual International Multimodal Conference (Multimedia '97), (Seattle, WA, November 1997), ACM Press, pp 31-40.
3. Cohn, A. (1996) Calculi for Qualitative Spatial Reasoning. In Artificial Intelligence and Symbolic Mathematical Computation, LNCS 1138, eds: J Calmet, J A Campbell, J Pfalzgraf, Springer Verlag, 124-143, 1996.
4. Cohn, A. Magee, D., Galata, A., Hogg, D., and Hazarika, S. (2002) Towards an Architecture for Cognitive Vision using Qualitative Spatio-Temporal Representations and Abduction. *Proc. Spatial Cognition 2002*, Tutzing Castle, Lake Starnberg, May 2002.
5. Coventry, K. 1998. Spatial prepositions, functional relations, and lexical specification. In Olivier, P. and Gapp, K.P. (Eds) 1998. *Representation and Processing of Spatial Expressions*. LEA Press.
6. Davies, J. and Goel, A. (2001) Visual analogy in problem solving. *Proceedings of IJCAI-01*.
7. Edwards, G. and Moulin, B. 1998. Toward the simulation of spatial mental images using the Voronoi model. In Olivier, P. and Gapp, K.P. (Eds) 1998. *Representation and Processing of Spatial Expressions*. LEA Press.
8. Egenhofer, M. (1997) Query Processing in Spatial-Query-by-Sketch in Journal of Visual Languages and Computing 8(4), 403-424 pp.
9. Falkenhainer, B., Forbus, K., Gentner, D. (1989) The Structure-Mapping Engine: Algorithm and examples. *Artificial Intelligence*, 41, pp 1-63.
10. Feist, M. and Gentner, D. 1998. On Plates, Bowls and Dishes: Factors in the Use of English IN and ON. *Proceedings of the 20th annual meeting of the Cognitive Science Society*
11. Ferguson, R. W. 2000. Modeling orientation effects in symmetry detection: The role of visual structure, Proceedings of the 22nd Conference of the Cognitive Science Society . Hillsdale, New Jersey: Erlbaum.

12. Forbus, K., Ferguson, R. and Usher, J. 2001. Towards a computational model of sketching. *IUI'01*, January 14-17, 2001, Santa Fe, New Mexico
13. Forbus, K., Gentner, D. and Law, K. 1995. MAC/FAC: A model of Similarity-based Retrieval. *Cognitive Science*, 19(2), April-June, pp 141-205.
14. Forbus, K., Nielsen, P. and Faltings, B. "Qualitative Spatial Reasoning: The CLOCK Project", *Artificial Intelligence*, **51** (1-3), October, 1991.
15. Forbus, K. and Usher, J. 2002. Sketching for knowledge capture: A progress report. *IUI'02*, January 13-16, 2002, San Francisco, California.
16. Forbus, K., Usher, J., and Chapman, V. 2003. Sketching for Military Courses of Action Diagrams. *Proceedings of IUI'03*, January, 2003. Miami, Florida.
17. Gentner, D., & Markman, A. B. (1997). Structure mapping in analogy and similarity. *American Psychologist, 52,* 45-56.
18. Glasgow, J., Chandrasekaran, B., & Narayanan, N. (Eds.). (1995). *Diagrammatic Reasoning*. Cambridge Mass.: AAAI Press/The MIT Press.
19. Gross, M. and Do, E. (1995) Drawing Analogies - Supporting Creative Architectural Design with Visual References. in *3d International Conference on Computational Models of Creative Design*, M-L Maher and J. Gero (eds), Sydney: University of Sydney, 37-58.
20. Kuipers, B. 2000. The Spatial Semantic Hierarchy. *Artificial Intelligence* **119**: 191-233.
21. Leake, D. (Ed.) 1996. Case-Based Reasoning: Experiences, Lessons, and Future Directions, MIT Press.
22. Marr, D. 1983. *Vision: A computational investigation into the human representation and processing of visual information*. W. H. Freeman & Co.
23. Palmer, S. (1999) *Vision Science: Photons to Phenomenology*. MIT Press.
24. Rasch, R., Kott, Al, and Forbus, K. 2002. AI on the Battlefield: An experimental exploration. *Proceedings of the 14th Innovative Applications of Artificial Intelligence Conference*, July, Edmonton, Canada.
25. Stahovich, T. F., Davis, R., and Shrobe, H., "Generating Multiple New Designs from a Sketch," in Proceedings Thirteenth National Conference on Artificial Intelligence, AAAI-96, pp. 1022-29, 1996.