Open-domain sketch understanding: The nuSketch approach

Kenneth D. Forbus, Kate Lockwood, Matthew Klenk, Emmett Tomai, and Jeffrey Usher

Qualitative Reasoning Group, Northwestern University
1890 Maple Avenue, Evanston, IL, USA
Contact: forbus@northwestern.edu

Abstract

Sketching is often used when working out ideas. This combination of drawing and conceptual labeling is a very natural and effective form of communication and problem-solving. Creating software that can participate in sketching provides many challenges. We outline the nuSketch approach to sketch understanding, which focuses on visual and conceptual understanding instead of recognition. We summarize three experiments in progress with the *sketching Knowledge Entry Associate* (sKEA), the first open-domain sketch understanding system. sKEA exploits a variety of visual and qualitative spatial reasoning capabilities and human-like analogical matching to tackle a variety of tasks. We present experimental results, and outline future plans.

The nuSketch approach

Sketching is a form of multimodal interaction where participants use a combination of interactive drawing and language to provide high-bandwidth communication. This communication relies on shared understanding, both visual and conceptual. Multimodal interface research has mainly focused on providing more natural interfaces to legacy software, using recognition technologies to provide the desired interaction (Alvarado and Davis 2001, Cohen et al. 1997). While such systems have been shown to be quite useful in practice, we take a radically different approach. The nuSketch approach is based on a key observation about human-to-human sketching: Recognition is not essential. People are not artists in real time; they rely on language for conceptual labeling much of the time. While some specialized domains have visual symbol languages that practitioners use fluently, sketching is used far more broadly than that. In other words, for people, recognition is an accelerant, not a necessity.

This suggests a very different approach to sketch understanding that complements recognition-oriented research. In the nuSketch approach, we sidestep recognition issues by providing other interface mechanisms for people to conceptually label their ink. This provides two crucial advantages. First, it enables us to focus on deeper visual and conceptual understanding of sketches. Second, it enables us to build sketching systems that can operate outside the tight domain constraints that bind today's recognition-based systems. In particular, the sketching Knowledge Entry Associate (sKEA) (Forbus and Usher 2002) is, we believe, the first open-domain sketch understanding system. sKEA's only coverage limitation comes from the underlying knowledge base it uses – currently a subset of Cyc, consisting of over 35,000 concepts, constrained by 1.2M facts. This does not mean that we can effectively reason with all of these concepts in sketches yet – that is the nature of the challenge we have undertaken!

The nuSketch Architecture and sKEA

In the nuSketch architecture, the basic unit in a sketch is a glyph. Every glyph has *ink* and *content*. The ink consists of one or more poly-lines, representing what the user drew. The content is a conceptual entity, the kind of thing that the glyph is representing. There are two key problems that any sketching system must solve:

1. *Segmentation*. What pieces of ink should be considered together as glyphs?
2. *Conceptual Labeling*. What conceptual entity does a glyph denote?

We solve the segmentation problem by having the user click a button when they begin drawing a glyph and click it again when they are finished. Other segmentation techniques, such as time-outs and connectivity, are in our experience very frustrating for users and error-prone. We solve the conceptual labeling problem by enabling the content of a glyph to be declared, via a simple dialog, as one or more types drawn from the underlying knowledge base. This requires users to be familiar with the subset of the knowledge base that they need in their task, which is a strong limitation with the current version of the system. (We return to this issue at the end.)

The nuSketch architecture uses a variety of geometric computations to visually construct qualitative representations, including RCC8 relations (Cohen 1996), Voronoi diagrams (Edwards and Moulin 1998) for approximating proximity, and polygon operations to capture domain constraints.

A central feature of nuSketch is our use of analogical processing, based on Gentner's structure-mapping theory (Gentner 1983). Analogy provides a powerful means of entering and testing knowledge. Currently sKEA enables users to compare two layers of a sketch, which enables the detection of similarities and differences. We use the Structure-Mapping Engine (SME) (Falkenhainer, Forbus and Gentner 1989) to perform the comparisons. SME is a general-purpose analogical matcher. sKEA's analogies are based on both the visual and the conceptual material in a sketch. Some of our experiments also take advantage of the MAC/FAC system (Forbus, Gentner and Law 1994) which does a coarse matching using content vectors and then uses SME to narrow the results.

SME produces candidate inferences, conjectures about one description based on its alignment with another. Candidate inferences are useful in knowledge capture because they suggest ways to flesh out a description based on similarities with prior knowledge.

Since there is independent psychological evidence that structural alignment occurs in visual processing, and that SME captures many aspects of this processing accurately, it means that when our sense of similarity and our software's sense of similarity about a sketch diverge, it is a sign that our representations have failed to capture something crucial about the sketch. This provides a powerful constraint that drives the visual and conceptual representations we compute. Even if one does not care about modeling human performance, a sketching system will be a better partner if you and it agree on when things look alike.

Spatial Processing in nuSketch and sKEA

While we do not use recognition techniques on our sketches, we do compute some simple spatial properties of the ink (Forbus and Usher 2002). We focus on the spatial relationships between the glyphs rather than doing detailed analysis of the structure of the glyphs themselves; we call this approach *blob semantics*. When a glyph is added, moved, or resized, sKEA computes a set of spatial attributes and relationships. This process is described in detail in (Forbus, Tomai, and Usher 2003). The spatial attributes and relationships that make up the visual structure of the sketch are: groupings, positional relationships, size, and orientation.

sKEA automatically computes two kinds of groupings: *contained glyph groups* and *connected glyph groups*. A contained group consists of a single container glyph and the set of glyphs that are fully contained within it, possibly tangentially so. The contained group does not include glyphs that are contained within other glyphs in the group. A connected glyph group consists of a set of glyphs that overlap ink strokes with one another. Articulation points can be computed over connected glyph groups and tangentially connected pairs of glyphs can be noted as such.

Positional relationships are computed pair-wise and expressed in a viewer-oriented coordinate system of left/right and above/below. They are not computed between all pairs of glyphs, but rather in local neighborhoods based on adjacency, as determined via a Voronoi diagram. Positional relationships are computed only between glyphs on the same layer of a sketch.

Glyph size in sKEA is assigned as either tiny, small, medium, large or huge. Sizes are based on the area of a glyph's axis-aligned bounding box, a coarse but empirically useful approximation. Glyph areas are normalized with respect to either the area of the bounding box around all glyphs on all layers or the users view port, whichever is larger. The normalized areas are then clustered into qualitative size values based on a logarithmic scale of the square root of the area.

Additional Spatial Reasoning

As part of our ongoing research we are developing more spatial reasoning capabilities to accomplish different research goals. Currently we are working on adding the ability to articulate the "important" points and segments on different objects as well as to more specifically define the relationship between adjacent glyphs. Another area where we need to expand our available spatial reasoning abilities is curvature. We are working on different techniques to qualitative summarize degree of curvature.

Experiments

We next describe three experiments currently underway with sKEA. All three leverage sKEA as an input device to spatial reasoning systems as this is an area that is well-suited to sketching as input. Our qualitative spatial reasoning approach provides a bridge between the perceptual and the conceptual.

Experiment 1: Miller Geometric Analogies

Evans' ANALOGY program is an AI classic (Evans 1968). It seems only natural that sKEA ought to be able to carry out this task. And our recent results suggest that in fact it can. Here is a typical Miller problem, drawn using sKEA: The test-taker must choose one of the five answers as providing the closest analog to the comparison "A is to B as C is to *blank.*" Evans system solved these problems by constructing an explicit transformation that turned A into B, computed transformations between C and 1, C and 2 and so on, and chose the transformation from C that is the closest to the A to B transformation. As Evans noted, there can be ambiguity in the appropriate choice of transformation.

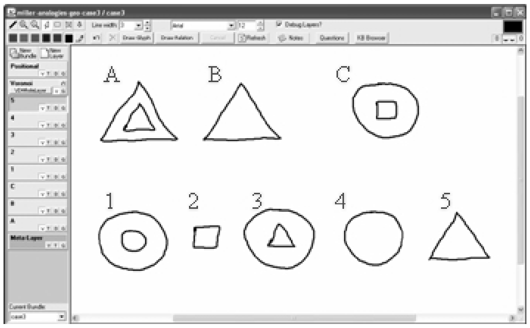

Figure 1. A typical Miller test problem

We avoid this problem by simply using SME to provide human-like analogical processing, comparing the similarities and differences directly instead of constructing transformations. Because SME is domain independent, we are able to focus our investigation on the representation of the problems.

To solve the geometric analogy problems, we use a two-stage structure mapping process, depicted in figure 2 below. The first stage is the computation of mappings from figure A to figure B and from figure C to each of the answer figures 1-5. This generates six mappings (the example mapping AB and the potential answer mappings C1-C5) that represent the similarities and differences between their respective pairs of figures. The second stage takes those mappings as input and computes the prescribed analogy from AB to each of the answer mappings C1-C5. The strongest results from the second stage indicate the correct answer. The second stage is an example of what we call *second order analogical mapping*.

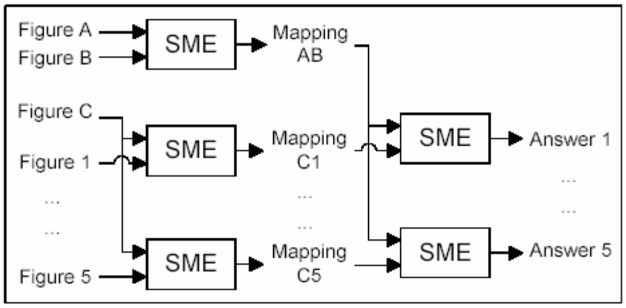

Figure 2. The two-stage mapping process

Since we do not have built-in recognition of simple shapes, we currently have to use conceptual labeling to identify the shapes to sKEA[1]. This strikes us as a natural opportunity to incorporate some simple recognition technologies, which we are planning to do.

In order to make meaningful comparisons between terms such as `Circle` versus `Triangle`, the system requires common-sense domain knowledge about those terms. It must know that circles, square and triangles are all types of shape and have the same kind of knowledge about sizes and orientations. This taxonomic information is contained in our knowledge base as Cyc `genls` and `isa` relationships. In comparing sizes, there is additional ordering information from smallest to largest, and in comparing orientations there is the concept of rotation from one orientation to the next. These facts form the domain of knowledge necessary for the solution of these geometric analogy problems. We make the knowledge available to the system in the form of general knowledge within our knowledge base.

The system elaborates the results of each first-stage mapping by querying the knowledge base, retrieving knowledge based on the attributes in the mapping and what relationships hold between them. These elaborated descriptions become input for the second stage of analogical mapping.

Experimental Results

When the Miller Geometric analogy test is run through our system, it scores correctly on 15 of the problems, incorrectly on 4 and gives ambiguous results on one problem. The inability of our system to solve all of the problems can be traced to four short-comings. They are (1) the inability to do axial symmetry, (2) the inability to decompose glyphs (due to the blob semantics assumption), (3) a lack of hierarchical awareness in positional relationships, and (4) the inability to reinterpret the example pair and try a different avenue of attack. For a more detailed analysis of these issues please see (Tomai, Forbus and Usher 2004).

Future work in this area will include continued research on visual structure as well as conceptual relationships. We plan to extend our visual processing and experiment with Ferguson's MAGI model of symmetry (Ferguson 1994). We also intend to introduce conceptual grouping as both context for spatial qualities and as a foundation for richer conceptual relationships between sketched entities. Finally, we plan to stretch the boundaries of blob semantics by exploring automatic recognition of known shapes and techniques for the decomposition of blobs into visually meaningful pieces of ink. While useful to this task, all of these ongoing improvements are completely domain neutral and will contribute to other research areas as well.

Experiment 2: Bennett Mechanical Comprehension test

The Bennett Mechanical Comprehension test has been used for over 50 years as a method for evaluating candidates for jobs requiring mechanical aptitude. It is

[1] Evans had a preprocessor that was run for half of his examples to automatically construct shape descriptions. The rest were input by hand.

also commonly used as an independent measure of spatial ability by cognitive psychologists. We are using this test as one means of evaluating the physical knowledge and reasoning skills in Companion Cognitive Systems (Forbus and Hinrichs 2004), a new architecture we are creating. Each problem involves an analysis of a picture to understand the question and arrive at the proper answer. In addition to providing an externally determined evaluation metric, this test is especially good because it is extraordinarily broad in terms of its domain content. Having a small set of principles isn't enough, knowing how those principles are applied to real-world situations is also crucial.

The test consists of 68 problems, including statics, dynamics, fluid, thermal, electricity, and materials. The problems are all qualitative in nature. Consider for example the two ladders shown here. Which would be more stable, A or B? This is a simple kind of comparative analysis problem – once one has mapped from the everyday concepts to abstract qualitative mechanics! It is how to understand everyday objects in terms of qualitative mechanics that is an interesting learning problem. For example, conceptual properties such as "stability" must be tied to visual properties like "the width of the base", which in turn are grounded in the sketch. Being able to learn these visual/conceptual mappings and use them by analogy to solve new problems is our goal.

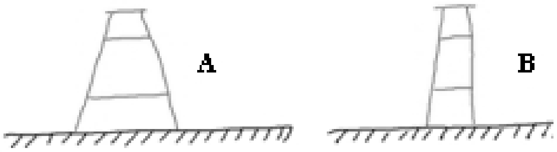

Figure 3. Ladder stability problem.

One important subproblem in analogical reasoning using sketches is mapping properties such as measurements from one sketch to another, as in the width of the base in Figure 3. Our solution is to specify measured properties in terms of *anchor points* within a glyph that are easily distinguishable in qualitative terms. Here, for example, the bottom points of the glyph are key to defining the width of the base. By examining all of the points on the glyph we can find the bottom points, including the leftmost and rightmost bottom points. Such anchor points are computed on demand. Thus terms like "width of the base" can be defined for one ladder in terms of a symbolic expression, and applied by analogy to other situations.

The set of anchor points we have needed so far consists of the following. On individual glyphs, we can compute leftmost bottom, rightmost bottom, leftmost top, rightmost top, bottom leftmost, bottom rightmost, top leftmost, top rightmost and centroid. On pairs of glyphs, we can compute intersection points and overlapping segments. We doubt that this exhausts the relevant set of distinctions needed, but we would be surprised if it were, say, three times this size. Any reasoning system can take advantage of these as sKEA provides access to these points in predicate calculus through NATs, such as (RightmostBottomFn (GlyphFn Object-13 Layer-12)), and as reified objects, such as Point-13.

Companions will use analogy over sketches heavily in solving these problems. When presented with a problem, a Companion will use MAC/FAC to search its case library of experiences for possible analogues. These analogues include sketches created in presenting prior problems and in "bootstrapping" knowledge entry sessions, where how concepts such as ladders and wheelbarrows will be depicted by multiple users drawing them in problem-independent ways. In the example of Figure 3, the case it would retrieve would include that "stability" is qualitatively proportional to "the width of the base". Then, via an analogy using SME, that fact would be assumed in the current case as a candidate inference and the sKEA would compute the measurements for "the width of the base" of the each ladder. To arrive at a solution, differential qualitative analysis (Weld 1988) has been implemented as a list of back chaining axioms in FIRE. The corresponding parts between the two systems are also found by analogy (i.e., comparing parts A and B of Figure 3). At this writing, we have implemented and tested all of the pieces individually, and by the time of the Symposium we will be able to report on how well they work together to solve problems.

Experiment 3: Modeling spatial language

Connections between space and language can be surprisingly subtle. For instance, many accounts of spatial prepositions only take geometry into account. However, there is ample psychological evidence that spatial prepositions rely on conceptual factors as well (Coventry 1998, Feist and Gentner 1998). For example (illustrated in figure 4a below), given an abstract blob on a curved surface, people are more likely to say that the blob is on it if the blob is described as animate (e.g., a dragonfly) and more likely to be in it if the curved surface is described to be a hand. Language can even affect spatial memory, as Feist & Gentner have found (Feist and Gentner 2001). If subjects are shown the picture on the right while being told "The puppet is on the table", when they are later shown both pictures they are more likely to claim that the picture they saw was the one on the left.

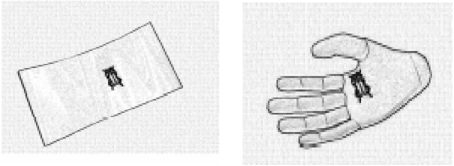

Figure 4a. Conceptual factors influence spatial prepositions

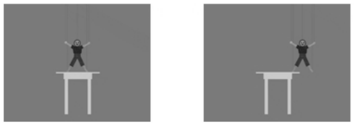

Figure 4b. Spatial prepositions influence memory

These psychological results suggest that accurately modeling human use of spatial prepositions requires reasonable conceptual and linguistic, as well as spatial, models. We are currently using sKEA to model these phenomena. We gather spatial information from the sketch ink. Functional information is gathered from the label the user assigns to the ink and the knowledge base. For example, if the user draws a figure like the one in Figure 4a above and labels the ground as `Plate` the functional properties can be gathered from Cyc by looking at the collections of which plate is a member (is it `Animate`, a `Container`, etc). This can be combined with the geometric analysis of the ink.

This process involves making visual ties to the spatial relationships represented inside the Cyc KB (which, for spatial prepositions, were already motivated by the cognitive science literature) and seeing if we can model these findings.

As this writing, we are just in the beginning stages of this particular project. However, it is another example of how we hope to integrate the power of open-domain sketching with cognitive systems to produce interesting and useful results.

Related Work

Most other existing multimodal interfaces focus on creating an extremely natural interaction using recognition techniques and other algorithms to automatically recognize user sketches. The tradeoff imposed is that they operate in a tightly constrained domain. sKEA on the other hand, can operate in arbitrary domains, the only limitations being the specificity of the underlying database and what is natural to express via sketching. The price we pay is a slightly less natural interaction between the user and the system. We think the nuSketch approach and sKEA provide a valuable complement to the usual recognition-based approaches used in multimodal interfaces. To be sure, as recognition technologies improve we will happily incorporate them into nuSketch systems – as long as we can do so without compromising our open-domain approach.

Our use of SME, a general-purpose analogical matcher, for both visual and spatial representations is unique among approaches to analogy. Most attempts to build analogy systems have been domain-specific. For example, Mitchell's Copycat program (Mitchell 1993) is designed got use with letter strings, and French's TableTop (French 1995) woks only with table settings. The kinds of comparisons that can be made with these systems are hard-wired. Unfortunately, many case-based reasoning systems are similarly fixed in terms of their capabilities. Our experience with sKEA provides yet more evidence that this needn't be the case: Domain-independent matchers grounded in principles of human processing, like SME, can operate in a wide variety of domains.

Discussion and Future Work

We have presented three ongoing research projects related to our nuSketch and sKEA systems. Our domain-independent approach to sketch understanding allows us to build utilities for general purpose qualitative spatial reasoning and apply them to problems in different domains. The individual domains are restricted only by the contents of the knowledge base. The knowledge base is easily extendable through the use of flat-files so new domains are easily incorporated.

Our ongoing work has also pointed out weaknesses of our current spatial reasoning techniques. Work is currently underway to add additional functionality and to enhance what we already have. Several areas that were mentioned as additions in progress are: location and use of relevant points on a glyph, estimation of curvature, segmentation of glyphs, and expansion of the spatial vocabulary. We are also investigating using natural language to reduce the need for users to be intimately familiar with the formal details of a large knowledge base; this is a very complex undertaking, as the investigation of spatial prepositions above indicates.

References

1. Alvarado, Christine and Davis, Randall (2001). Resolving ambiguities to create a natural sketch based interface. *Proceedings of IJCAI-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. Cohen, 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.
4. Coventry, K. (1998). Spatial prepositions, functional relations, and lexical specification. In Oliver, P. and Gapp, K. P. (Eds). *Representation and Processing of Spatial Expressions.* LEA Press.
5. Edwards, G. and Moulin, B. (1998). Toward the simulation of spatial mental images using the Voronoi model. In Oliver, p. and Gapp, K. P. (Eds). *Representation and Processing of Spatial Expressions*. LEA Press.
6. Evans, T. (1968). A Program for the Solution of a Class of Geometric-Analogy Intelligence-Test Questions, *Semantic Information Processing*, 1968, MIT Press.
7. Falkenhainer, B., Forbus, K., and Gentner, D. (1989). The Structure-Mapping Engine: Algorithm and examples. *Artificial Intelligence*, 41, pp 1-63.
8. 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.*
9. Feist, M. I., & Gentner, D. (2001). An influence of spatial language on recognition memory for spatial scenes. *Proceedings of the Twenty-third Annual Conference of the Cognitive Science Society*, 279-284.
10. Ferguson, R. W. (1994). MAGI:Analogy-based encoding using symmetry and regularity, *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society.*
11. Forbus, K., Gentner, D., and Law, K. (1994). MAC/FAC: A Model of Similarity-based Retrieval. *Cognitive Science* 19, 141-205.
12. Forbus, K. and Hinrichs, T. (2004) Companion Cognitive Systems: A step towards human-level AI. To appear in *Proceedings of the 2004 AAAI Fall Symposium on Achieving Human-level Intelligence through Integrated Systems and Research.*
13. Forbus, K., Tomai, E. and Usher, J. (2003). Qualitative Spatial Reasoning for Visual Grouping in Sketches. *Proceedings of the 17th International Workshop on Qualitative Reasoning*, Brasilia, Brazil, August 2003.
14. Forbus, K. and Usher, J. (2002). Sketching for knowledge capture: A progress report. IUI'02, January 13- 16, 2002, San Francisco, California.
15. French, R. (1995). *The subtlety of similarity.* Cambridge, MA: The MIT Press.
16. Gentner, D. 1983. Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7, 155-170.
17. Mitchell, M. (1993). *Analogy-making as perception: A computer model*. Cambridge, MA: The MIT Press.
18. Tomai, E, Forbus, K. and Usher, J. (2004). Qualitative Spatial Reasoning for Geometric Analogies.
19. Weld, D. (1988). Comparative Analysis. *Artificial Intelligence*. 333-374.