

# Open-domain sketch understanding: The nuSketch approach

Kenneth D. Forbus, Kate Lockwood, Matthew Klenk, Emmett Tomai, Andrew Lovett  
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 utilizes a number of visual and qualitative spatial reasoning capabilities and human-like analogical matching to tackle a variety of tasks. We summarize some experimental results and outline future plans.

**Keywords:** Sketch understanding, visual reasoning, analogy

## The nuSketch approach

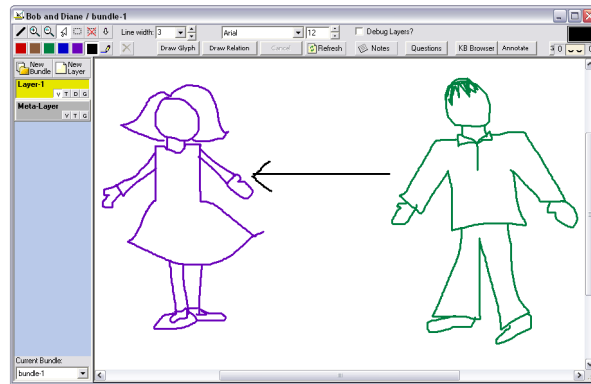
Sketching is a form of multimodal interaction where participants use a combination of interactive drawing and language to provide high-bandwidth communication. This communication relies on shared understanding, both visual and conceptual. Multimodal interface research has mainly focused on providing more natural interfaces to legacy software, using recognition technologies to provide the desired interaction [2, 4]. While such systems have been shown to be quite useful in practice, they are restricted to narrow domains of discourse and are subject to the limitations of current statistical recognition techniques. We take a radically different yet complementary approach.

The nuSketch approach is based on a key observation about human-to-human sketching: recognition is not essential. In sketching, the drawing provides the spatial aspects of what is being communicated while linguistic interaction provides interpretation of what is drawn. 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.

Many other researchers in this field are also looking at using language and gesture in combination with sketching to ease the burden on ink recognition [1, 21]. However, many of these systems rely on statistical speech or text recognizers which are

unfortunately susceptible to the same problems as ink recognizers. While the combination of recognition techniques appears to be less error prone than either alone, they are still relatively high.

We have taken 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 the constituents of their sketches. 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) [18] 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 1.2 million 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!



**Figure 1.** A screen shot of a typical sKEA sketch

## 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 grouped 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. Research has also shown that users often do not use continuous strokes when drawing an object, and may draw more than one object in a single stroke, making stroke-based methods error prone [25]. Successes with methods like pen speed are often restricted to very tight domains of symbols [26]. As these techniques improve we would be interested in adding them as another tool for visually analyzing glyphs. We solve the conceptual labeling problem by enabling the content of a glyph to be declared, via a simple dialog, as being instances of 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 limitation of the current version of the system. (We return to this issue at the end.)

In addition to drawing glyphs representing objects, users can draw glyphs representing the relationships between objects. These relationships are represented by arrows and are labeled using the underlying knowledge base. Currently arrows are the only object that sKEA recognizes. The direction of the arrow is used to fit the objects depicted by the surrounding glyphs to the roles in the relationship. Figure 1 above shows the use of a relationship arrow in a sketch.

SKEA gives users several tools to organize their sketches as well. Sketches can be divided into *bundles* and *layers*. Bundles represent coherent subsets of a sketch. This is based on the observation that when people sketch for communication, they naturally divide their work into *subsketches*. Subsketches are used to indicate local neighborhoods of ideas and in human-to-human sketching are often delineated by the context. Bundles are used in sKEA to provide this same division of sketches and are useful for tasks like breaking out the steps in a process or describing a system from different perspectives or at different levels of detail. Just as users can define relationships between glyphs, relationships between bundles can also be described. This is done by using a special view, the *metalayer*, from which bundles appear to be glyphs, and thus can be related by drawing arrows, just as they would within a bundle. Layers can be thought of as transparent sheets stacked on top of each other within a bundle. Layers give the user a way to divide a bundle into conceptual parts for reasoning.

The nuSketch architecture uses a variety of geometric computations to visually construct qualitative representations, including RCC8 relations [5], Voronoi diagrams [7] for approximating proximity, and polygon operations to capture domain constraints. These geometric computations are based on the visual properties of the ink in a sketch.

A central feature of nuSketch is our use of analogical processing, based on Gentner's structure-mapping theory [20]. The intuition of structure-mapping theory is that an analogy is the mapping of information from one structured description (the

base) to another (the target). This mapping is done through an incremental alignment process. The target objects do not need to resemble their corresponding base objects, they are placed into correspondence by virtue of the roles they play in the relational structure of the cases.

Analogy provides a powerful means of entering and testing knowledge. Currently sKEA allows users to compare two bundles of a sketch or two bundles from different sketches, which enables the detection of similarities and differences. We use the Structure-Mapping Engine (SME) [9] 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 since it is assumed that people would use similar glyphs to describe similar concepts. Some of our experiments also take advantage of the MAC/FAC system [15], which simulates similarity-based reminding. MAC/FAC uses a first stage of 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. As described below, systems that use sKEA tend to use candidate inferences heavily in their reasoning. Candidate inferences are useful when reasoning about sketches because sketches tend to portray concrete examples. People find it easier to express knowledge about concrete, particular situations, which is one way that sketching simplifies knowledge capture. By exploiting analogy in reasoning with sketches, we can directly exploit knowledge expressed about particulars in sketches to reason about new situations.

Since there is independent psychological evidence that structural alignment occurs in visual processing, and that SME captures many aspects of similarity processing accurately, we know 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. We call this the *shared similarity constraint* and it is central to many of our on-going projects.

## 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 [17]. 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 [18]. 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. For example, in Figure 2, the eyes, nose and mouth form a contained glyph group with the head as the container. 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. In Figure 2, the body, legs, tail, and head of the cat constitute a connected glyph group. Articulation points can be computed over connected glyph groups and tangentially connected pairs of glyphs can be noted as such.



**Figure 2.** A sketch with contained and connected glyph groups

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 [7]. Positional relationships are computed only between glyphs on the same layer of a bundle.

Glyph size in sKEA is qualitative and is assigned as `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 user's 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.

Users may also want to specify the conceptual relationships that hold between entities in their sketch, based on the visual relationships that hold between the glyphs that represent them. For example, two glyphs that touch visually may conceptually be connected by a hinge, bonded together, or simply touching. We call these relationships *visual/conceptual relationships* since they combine both visual and conceptual knowledge. sKEA provides tools for users to specify these relationships so that knowledge can be stored as part of the sketch.

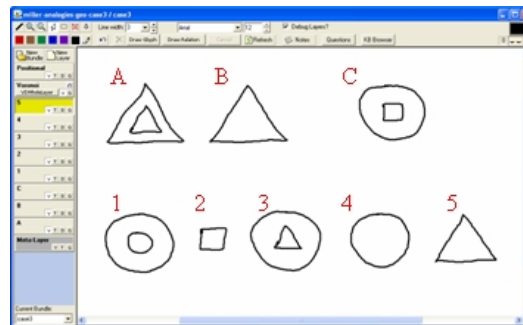
## Projects

We next describe three projects 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 [13]. We also indicate areas where our experiments have highlighted short-comings of the nuSketch architecture and outline our plans to add functionality to address these issues.

### Project 1: Miller Geometric Analogies

Evans' ANALOGY program, which solved problems from the Miller Geometric Analogies Test, is an AI classic [8]. It seems only natural that sKEA ought to be able to carry out this task. Our recent results demonstrate that it can. Figure 3 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

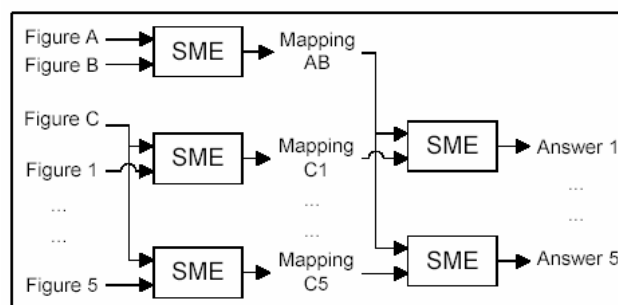
transformation that turned A into B, computed transformations between C and 1, C and 2 and so on, and chose the transformation from C that is the closest to the A to B transformation. As Evans noted, there can be ambiguity in the appropriate choice of transformation.



**Figure 3.** 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 4 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 4.** 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<sup>1</sup>. This strikes us as a natural opportunity to incorporate some simple recognition technologies, which we are planning to do.

Typically sKEA does not decompose glyphs, due to the blob semantics model it uses. We found it useful for these problems to introduce the ability to decompose glyphs into edges, to provide a means for the system to recognize that one shape was a rotation of another. Each edge is a maximal series of connected points that can be distinguished from the rest of the glyph because the edge either is disconnected from the other points or adjoins another edge at a saliently sharp corner. Once edges have been determined, the system finds all the corners where edges are connected. The system can then find any closed shapes in the glyph. A closed shape consists of a set of adjoining edges in which the first edge is the same as the last. The system builds an orientation-independent representation of the most significant closed shape found in a glyph. SME is used with these orientation-independent representations for pairs of glyphs. If a likely match is found (a match in which every edge in one glyph has a corresponding edge in the other glyph) then the system actually attempts the rotation. It looks at the difference in orientation for each pair. If all of the differences are relatively close to each other, as determined by a threshold, the system concludes that one glyph is the rotation of the other. It can approximate the degree of rotation between the glyphs by averaging the degree of rotation between each edge pair.

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 shapes 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 necessary facts are available to the system as part of its knowledge base.

---

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



The system elaborates the results of each first-stage mapping by querying the knowledge base and 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 on Miller test problems**

Our system does quite well on the subset of the Miller Geometric Analogy Test questions used by Evans. It scores correctly on 16 of the problems, scores incorrectly on three and gives ambiguous results on one problem. The inability of our system to solve all of the problems can be traced to three short-comings. They are:

1. The inability to do axial symmetry,
2. A lack of hierarchical awareness in positional relationships
3. The inability to reinterpret the example pair and try a different avenue of attack.

For a more detailed analysis of these issues please see [16]<sup>2</sup>. We plan to extend our visual processing with Ferguson's MAGI model of symmetry [12], to handle the first problem. To address the second problem, we intend to introduce conceptual grouping as both context for spatial qualities and as a foundation for richer conceptual relationships between sketched entities. The third problem suggests that a more flexible processing architecture may be needed in some cases.

The Miller problem illustrates two boundaries on the utility of blob semantics. The first, which we overcame by decomposing glyphs into edges, is the need to understand the shape of a blob well enough to recognize when two blobs represent the same shape but rotated. The second limitation is our need to manually label simple shapes. We are currently exploring automatic recognition methods to handle simple shapes.

### **Project 2: Bennett Mechanical Comprehension test**

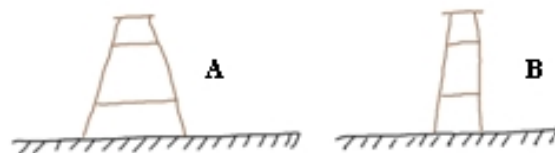
The Bennett Mechanical Comprehension test has been used for over 50 years as a method for evaluating candidates for jobs requiring mechanical aptitude. It is also commonly used as an independent measure of spatial ability by cognitive psychologists. We are using this test as one means of evaluating the physical knowledge and reasoning skills in Companion Cognitive Systems [16], a new cognitive 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,

---

<sup>2</sup> When [27] was written, the solution to rotated figure recognition had not yet been developed.

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 5.** The 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 5. 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. These are especially useful because our implementation of qualitative mechanics on rigid bodies, which currently covers the analysis of lever and force distribution systems, talks about surfaces that we interpret spatially as the overlapping parts of glyphs. 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 non-atomic terms, such as (RightmostBottomFn (GlyphFn Object-13 Layer-12)).

The Companions architecture is agent-based, with sKEA being one of the agents it can use. Companions use analogy over sketches heavily in solving these problems. When presented with a problem, a Companion uses MAC/FAC to search its case library of experiences for possible analogues. These analogues include sketches created in “bootstrapping” knowledge entry sessions, where how concepts such as ladders and wheelbarrows will be depicted by users drawing them in problem-independent ways. In the example of Figure 5, the case it retrieves includes that “stability” is qualitatively proportional to “the width of the base”. Then, via an analogy (using SME), that qualitative proportionality would be assumed in the current case as a candidate inference and sKEA would compute the measurements for “the width of the base” of the each ladder. To arrive at a solution, differential qualitative analysis [28] is used, where the corresponding parts between the two systems are also found by analogy (i.e., comparing parts A and B of Figure 3). Our first experiment based on these ideas is described next.

## **Method.**

For our initial experiment, we selected a set of 13 problems from the Bennett Mechanical Comprehension test. We developed, in text form, a list of 18 example situations, 15 of which were intended to be good analogues for specific test questions. We then had three users, all graduate students with varying degrees of familiarity with sKEA, draw each of the situations on the list. In each sketch the users sketched glyphs to describe the structure of the system. They used sKEA’s visual/conceptual relationship interface to fill in their intended relationships between the parts (e.g., two glyphs that are touching might, depending on the situation, represent objects that are bolted to one another or are free to slide over each other). When appropriate, they annotated the sketch with visual and conceptual quantities (e.g., the smoothness of ride for a vehicle), and causal laws linking them as appropriate. After they decided their sketch was complete, a qualitative reasoning system was run to formulate a model of the system depicted by the sketch. If they were satisfied with the model fragments found, they were done, otherwise they were encouraged to modify their sketch until they were satisfied with the model. For example, in a sketch depicting two gears meshed and one gear was annotated to be rotating, torque transfer should be identified as a possibility in the model.

The idea of the model formulation step is that the qualitative mechanics domain theory represents the Companion’s initial endowment of common sense knowledge. It is incomplete by design, containing only very simple models of how objects interact by pushes and pulls [22] – the rest we want to have the system learn. One of the most important things that the

system needs to learn is how to apply principles to real-world situations. By augmenting the examples with the idealizations used by the domain theory, the user-drawn sketches provide examples of how these idealizations work in real-world situations (as depicted by their sketches), and hence provide grist for applying to new situations via analogy.

To pose problems, the 13 test problems were drawn by a fourth graduate student and given to the system. This table illustrates how the number solved varied when using sketches from each of the users:

<b>Example Library</b>	<b>Number Correct</b>	<b>%</b>
<b>User1</b>	4	31%
<b>User2</b>	8	62%
<b>User3</b>	2	15%

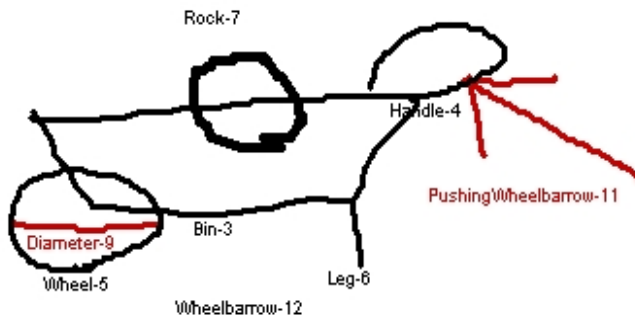
These results are, at best, mixed. User2's sketches were sufficiently close to those drawn by the problem enterer that the system was able to provide a respectable performance (statistically significant,  $P < 0.04$ ). Unfortunately, this was not true for the other two users (not statistically significant).

Analyzing how the system works sheds light on why this is happening. There are two main kinds of failures the system can exhibit. The first is retrieving a poor example. The second is not being able to use the example that it retrieved to solve the problem. The first-principles reasoning components of the system are based on well-worn qualitative reasoning techniques, so the responsibility must arise from a combination of analogical retrieval or analogical mapping (to apply the principles of the case). Analogical retrieval failures can be due to either representations that don't capture the commonalities in the situations, or examples drawn in very different ways from the test problems. Similarly, mappings can go awry either due to representation failures, or due to the retrieved example being extremely different.

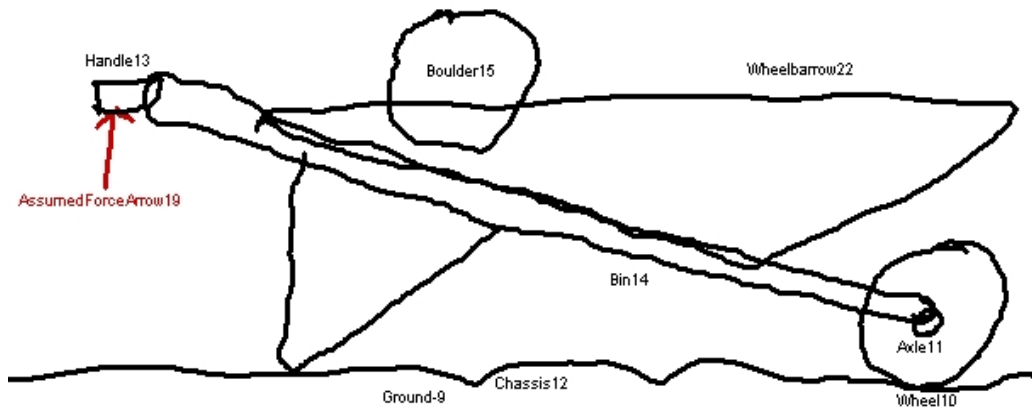
Scrutinizing the retrieval results sheds some light on this issue. Let us score a retrieval as "correct" if the example depicted was intended to be one of the analogues for that test problem.

<b>Example Library</b>	<b>Correct Retrievals</b>	<b>%</b>
<b>User1</b>	10	77%
<b>User2</b>	10	77%
<b>User3</b>	7	54%

These values are all statistically significant ( $P < 10^{-5}$ ). Our retrieval system, MAC/FAC, is clearly doing its job. A close analysis of the mappings in the failure cases indicates that there are two kinds of failures. The first kind is differences in the sketches that are so significant that it would be unreasonable to expect an analogy. (Remember, the users started with text describing the situation, often just a simple phrase or at most a couple of sentences.) The second kind is due to small differences in the representation that might be reasonable for a person to overcome. For example wheelbarrows occurred in two of the problems on the test. Figure 6a shows the example sketch and Figure 6b shows the problem sketch. Not only are there different numbers of glyphs in these sketches, but the entities that the glyphs depict are of different types. The example sketch does not contain a glyph for the ground or for the axle where the problem sketch does. Furthermore, the glyph in the example sketch that refers to the leg is conceptually labeled as a chassis in the problem sketch.



**Figure 6a.** Example Wheelbarrow



**Figure 6b.** Problem Wheelbarrow

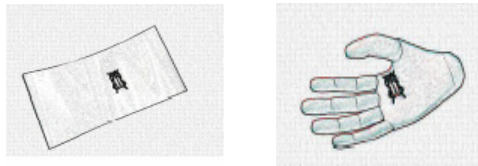
These results, in retrospect, are not too surprising. It is well-known that human problem solving is often very contextualized, with principles rarely applied in all of the circumstances in which they are appropriate [14]. One hallmark of expertise is the ability to encode situations in ways that make it easier to apply principles [3]. To get closer to human performance, our Companions need to be extended with human-like generalization techniques and ways to learn new encoding techniques.

### **Future work on the Bennett test**

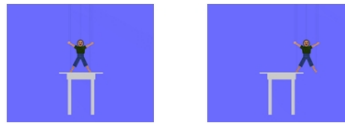
We are working on expanding the systems capabilities to tackle the entire test. So far we have dealt with force transfer between rigid bodies. To accomplish the entire test, the system must have a qualitative understanding of fluids, basic electricity and materials. Spatially, the system will have to reason about angles, flexible shapes, and how fluids interact with surfaces. Some of these representations have been worked out with symbolic inputs (cf. [22]), but tying them effectively to sketched input will require additional research. The spatial reasoning and representation extensions mentioned earlier in the context of the Miller Analogy Test will, we believe, also be useful in this test. And, as the results above indicated, incorporating something like our SEQL model of generalization [23] seems to be essential.

### **Project 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 [6, 10]. For example (illustrated in Figure 7a 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 [11]. If subjects are shown the picture on the right of Figure 7b 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 7a.** Conceptual factors influence spatial prepositions



**Figure 7b.** 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 7a 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). We hypothesize that this conceptual information can be combined with the geometric analysis of the ink to carry out human-like assignment of spatial prepositions. We are currently exploring this hypothesis by 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 use sKEA to 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 are integrating the power of open-domain sketching with cognitive systems to produce what we hope are interesting and useful results.

## Related Work

Most other existing multimodal interfaces focus on creating natural interaction using recognition techniques and other algorithms to automatically identify components of 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 coverage of

the underlying knowledge base 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 visual analogy. Most attempts to build analogy systems have been domain-specific. For example, Mitchell's Copycat program [24] is designed for use with letter strings, and French's TableTop [19] works 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 using sKEA, which is built upon the nuSketch architecture. 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, so new domains are easily incorporated.

Our ongoing work has also pointed out weaknesses in our current library of spatial reasoning techniques. Work is currently underway to expand these capabilities in several directions. For example, currently we are working on adding the ability to articulate the "important" points and segments on objects. We are working on different techniques to qualitatively summarize degree of curvature. We are also examining the constraints of blob semantics and looking at how to relax them to allow more segmentation of glyphs and hierarchical organization of glyphs in a sketch. We think the approach to segmentation and closed-shape recognition we developed for the Miller system is very promising, and we plan to migrate it into the nuSketch architecture in the near future. We also conjecture that this technique may be similar to one of the strategies people use for computing mental rotations, a conjecture we will explore through future cognitive simulation work.



We are also investigating using restricted natural language processing during conceptual labeling, to reduce the need for users to be intimately familiar with the formal details of a large knowledge base.

Our goal is that, ultimately, sKEA and its descendants will be able to interact as partners in sketching as flexibly and fluently as any human would.

## References

1. Adler, A., Eisenstein, J., Oltmans, M., Guttentag, L., and Davis, R. Building the Design Studio of the Future. *Proceedings of the AAAI Fall Symposium on Making Pen-Based Interaction Intelligent and Natural*. Washington, D.C. October, 2004.
2. Alvarado, Christine and Davis, Randall (2001). Resolving ambiguities to create a natural sketch based interface. *Proceedings of IJCAI-2001*.
3. Chi, M. T. H., Feltovich, P. J., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5, 121-152.
4. 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.
5. 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.
6. 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.
7. 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.
8. Evans, T. (1968). A Program for the Solution of a Class of Geometric-Analogy Intelligence-Test Questions, *Semantic Information Processing*, 1968, MIT Press.
9. Falkenhainer, B., Forbus, K., and 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 20<sup>th</sup> annual meeting of the Cognitive Science Society*.
11. 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.
12. Ferguson, R. W. (1994). MAGI:Analogy-based encoding using symmetry and regularity, *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*.
13. Forbus, K., Ferguson, R. and Usher, J. 2001. Towards a computational model of sketching. *IUI'01*, January 14-17, 2001, Santa Fe, New Mexico.
14. Forbus, K. and Gentner, D. 1997. Qualitative mental models: Simulations or memories? *Proceedings of the Eleventh International Workshop on Qualitative Reasoning*, Cortona, Italy.
15. Forbus, K., Gentner, D., and Law, K. (1994). MAC/FAC: A Model of Similarity-based Retrieval. *Cognitive Science* 19, 141-205.
16. 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*.
17. Forbus, K., Tomai, E. and Usher, J. (2003). Qualitative Spatial Reasoning for Visual Grouping in Sketches. *Proceedings of the 17<sup>th</sup> International Workshop on Qualitative Reasoning*, Brasilia, Brazil, August 2003.
18. Forbus, K. and Usher, J. (2002). Sketching for knowledge capture: A progress report. *IUI'02*, January 13- 16, 2002, San Francisco, California.
19. French, R. (1995). *The subtlety of similarity*. Cambridge, MA: The MIT Press.
20. Gentner, D. 1983. Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7, 155-170.

21. Kaiser, E. Dynamic New Vocabulary Enrollment through Handwriting and Speech in a Multimodal Scheduling Application. *Proceedings of the AAAI Fall Symposium on Making Pen-Based Interaction Intelligent and Natural*. Washington, D.C. October, 2004.
22. Kim, H. (1993). *Qualitative reasoning about fluids and mechanics*. Ph.D. dissertation and ILS Technical Report, Northwestern University.
23. Kuehne, S., Forbus, K., and Gentner, D. 2000. SEQL: Category learning as incremental abstraction using structure mapping. *Proceedings of CogSci-2000*. Philadelphia, PA.
24. Mitchell, M. (1993). *Analogy-making as perception: A computer model*. Cambridge, MA: The MIT Press.
25. Oltamns, M., Alvarado, C., and Davis, R. ETCHA Sketches: Lessons Learned from Collecting Sketch Data. *Proceedings of the AAAI Fall Symposium on Making Pen- Based Interaction Intelligent and Natural*. Washington, D.C. October, 2004.
26. Stahovich, T. Segmentation of Pen Strokes Using Pen Speed. *Proceedings of the AAAI Fall Symposium on Making Pen-Based Interaction Intelligent and Natural*. Washington, D.C. October, 2004.
27. Tomai, E, Forbus, K. and Usher, J. (2004). Qualitative Spatial Reasoning for Geometric Analogies.
28. Weld, D. (1988). Comparative Analysis. *Artificial Intelligence*. 333-374.