

Diagrams Based on Structural Object Perception

Pourang Irani
Faculty of Computer Science,
University of New Brunswick
Box 4400 Fredericton, NB
Canada E3B 2A3
(506) 453-4566
pirani@omg.unb.ca

Colin Ware
Faculty of Computer Science,
University of New Brunswick
Box 4400 Fredericton, NB
Canada E3B 2A3
(506) 453-4566
cware@omg.unb.ca

ABSTRACT

Most diagrams, particularly those used in software engineering, are line drawings consisting of nodes drawn as rectangles or circles, and edges drawn as lines linking them. In the present paper we review some of the literature on human perception to develop guidelines for effective diagram drawing. Particular attention is paid to structural object recognition theory. According to this theory as objects are perceived they are decomposed into 3D set of primitives called geons, together with the skeleton structure connecting them. We present a set of guidelines for drawing variations on node-link diagrams using geon-like primitives, and provide some examples. Results from three experiments are reported that evaluate 3D geon diagrams in comparison with 2D UML (Unified Modeling Language) diagrams. The first experiment measures the time and accuracy for a subject to recognize a sub-structure of a diagram represented either using geon primitives or UML primitives. The second and third experiments compare the accuracy of recalling geon vs. UML diagrams. The results of these experiments show that geon diagrams can be visually analyzed more rapidly, with fewer errors, and can be remembered better in comparison with equivalent UML diagrams.

Keywords

Perception, diagramming, geon theory, 3D diagrams, visual display, information visualization.

1. INTRODUCTION

Diagrams play an important role in the daily communication between humans, in our planning and problem solving. Recently there has been considerable interest in the theory that diagrams can act as cognitive "externalizations" enhancing cognition by mapping problem elements to a visual display in such a way that solutions become immediately evident [15][18]. However, not all mappings are equivalent. Human perception is based on a

sophisticated and complex information processing system designed to optimally extract environmental information. It can be argued that a diagram's effectiveness, to some extent, depends on how well it is designed as an input to this system [17][11].

In this paper we focus our attention on a common class of diagrams technically called graphs in computer science, and also sometimes called a node-link diagram. This category of diagrams includes software structure charts [14], entity relationship diagrams, and data flow models. Its essential characteristic is that of having two basic types of components, nodes and links or "edges". Nodes are used to represent a wide variety of entities, ranging from the extremely abstract to the concrete, and the links are used to represent many kinds of relationship including those that are structural, temporal, causal or functional [8]. There are many variations of node-link diagrams, but most commonly, the nodes are drawn as rectangular boxes, or circles, and the edges are lines or arrows that connect the nodes.

Graphs, as used by mathematicians and theoretical computer scientists are generally very abstract structures of nodes with linking edges. But for practical applications we need a somewhat more complex model. An elaborate form of graph diagram is the Universal Modeling Language (UML). This has its roots in both software engineering and databases and is designed for graphical modeling of complex systems [6]. As a generalization of the basic node-link graph, UML, and many other similar diagrams can be characterized as having,

- Heterogeneous nodes to represent a variety of different types of entities,
- Heterogeneous edges to represent a variety of different types of relationships,
- Attributes to both the entity nodes and the relationship edges.

An example of what we mean by attributes is the arcs in UML, which may have little symbols, or words attached to them giving additional information. For example a graph representing an information system might have differently shaped nodes representing server and workstation system components, and another shape of node representing individuals using the system. Links could represent customer relationships, supplier relationships, or communication channels between hardware components. Attributes might be defined for variables relating to the amount of business, the location of the components, the rate at which transactions take place, and so on.

The purpose of this paper is to propose a new way of presenting the kind of information we have been describing, based on structural object perception theory. We call it the "Geon Diagram". We begin by briefly reviewing the literature relating to perception of structural information. Following this we present a set of guidelines for constructing geon diagrams, and we present the results from three experiments that evaluate geon diagrams.

2. THEORIES OF OBJECT PERCEPTION

Modern cognitive theories of object perception can be divided roughly into two classes: the image-based and the structure-based approaches. Image-based theories emphasize the importance of the image plane in how visual information is stored and retrieved [16]. Thus, for example, an upside-down face is much harder to recognize than a right-side up face. Structure-based theories emphasize the way the visual system appears to extract the 3D structure of objects for recognition. For each theory, strong evidence has been given supporting their validity and it seems plausible that the visual system carries out both processes. In the following we concentrate on structural theories, especially Biederman's [2] geon theory, but we also recognize the importance of view dependent recognition and suggest that in drawing 3D diagrams special attention should be paid to their layout in the 2D plane.

2.1 Structure based object recognition

Marr and Nishihara developed and elaborated the structure-based approach to object recognition [13]. In their approach an object can be decomposed into a set of generalized cones. Thus for example, to differentiate between a horse and a giraffe, the decomposition would result in an arrangement of generalized cones of differing lengths approximating the head, tail, neck, limbs and body of the animals.

Silhouettes appear to be especially important in determining how we perceive the structure of objects. Halverston [7] noted that modern children tend to draw objects on the basis of the most salient silhouettes, as did early cave artists. Many objects have particular silhouettes that are easily recognizable; think of a teapot, a shoe, a church, a person or a violin. These canonical silhouettes are often based on a particular view of an object, often from a point at right angles to a major plane of symmetry. Marr [12] argued that "buried deep in our perceptual machinery" are mechanisms that contain constraints determining how silhouettes information is interpreted. There are three rules embedded in this perceptual machinery.

1. Each line of sight comprising a silhouette grazes the surface exactly once. The set of such points is the contour generator.
2. Nearby points on the contour of an image arise from nearby points on the contour generator of the viewed object.
3. All of the points on the contour generator lie on a single plane.

The idea of the contour generator is illustrated in Figure 1.

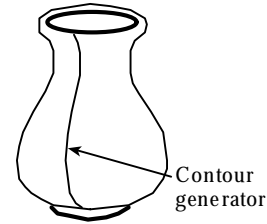


Figure 1. According to Marr the perceptual system makes assumptions that occluding contours are smoothly connected and lie in the same plane. Adapted from Marr [12].

Marr and Nishihara [13] suggested that concave sections of the silhouette contour are critical in defining how different solid parts are perceptually defined. Figure 2 illustrates a crudely drawn animal that we nevertheless readily segment into generalized cones representing the head body, neck, legs, etc. They also suggested a mechanism whereby the axes of the parts become cognitively connected to form a structural skeleton.

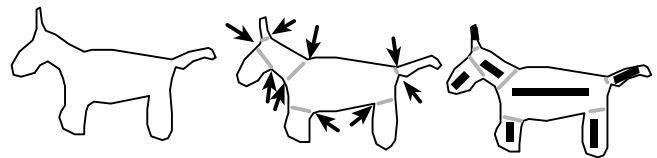


Figure 2. According to Marr and Nishihara [13] concave sections of the silhouette define sub-parts of the object. These points are critical in defining a structural skeleton. Adapted from Marr and Nishihara [13].

2.2 Geon Theory

The most elaborate theory of structural object recognition is the work of Biederman and his co-workers [2][3][4]. This theory proposes a hierarchical set of processing stages, arranged in layers, leading to object recognition (Figure 3). In the first two layers, information is decomposed into edges, then into component axes, oriented blobs, and vertices. At the intermediate level or layer 3, three-dimensional primitives such as cones, cylinders and boxes (called geons) are identified. In layer 4 the structure is extracted that specifies how the geon components interconnect; for example, for a human figure, the arm cylinder is attached near the top of the torso cylinder. Finally, object recognition is achieved. Hummel and Biederman [9].

A central concept in Biederman's approach is that a set of generalized cones or "geons" (short for geometrical ions) are 3D perceptual primitives. A family of 36 geons are defined by image properties on the silhouette contours in the 2D plane, by collinearity, symmetry, parallelism, curvature, and co-termination (the contours meet at a point, eg. a cone) [2].

Object identification also involves discovering relationships between the componential objects [1][2]. The decomposition of an object results in a geon structural description (GSD),

consisting of geons, their attributes, and their relations with adjacent geons. It is this structural description that contributes to viewpoint invariance, i.e. if two views of an object result in a similar GSD, then they should be treated as equivalent by the object recognition system. An overview of the entire object recognition process is given in Figure 3.

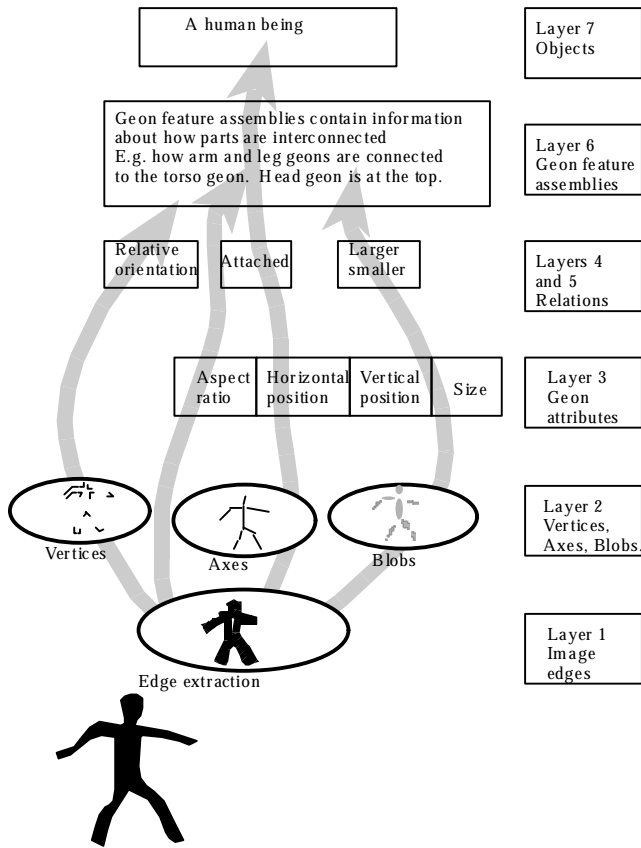


Figure 3. Biederman's theory of structural object perception proposes a series of processing stages, culminating in object recognition.

Different connections of the same geons can lead to different objects (Figure 4). A set of relations between geons can be defined in order to describe an object. Biederman has identified the following set of relationships as being particularly significant. A verticality relation between geons gives a visual indication as to the spatial organization between two geons, i.e. is geon A "on-top-of", "to-the-bottom-of", "at-the-right-of" geon B. The relative size of surfaces at join is another relation that determines whether a geon is connected to the shorter or longer surface of another geon. Off course this relation would not apply to all pairs of geons such as a sphere or a cube that do not have differing sizes of surfaces.

According to geon theory, color and texture are surface properties of geons that play a secondary role in perceptual object classification. These properties may aid in the recognition process, but do not constitute the defining characteristics. This is

particularly true for classes of objects with a clearly defined component structure. For example a table with well-defined boundary elements can be recognized as a table regardless of its color or texture.

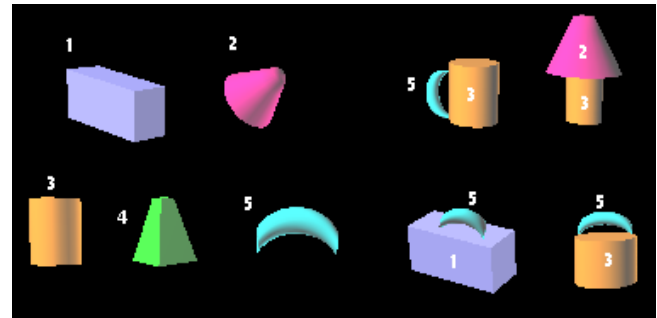


Figure 4. (a) Geons are object primitives in Biederman's theory. (b) When connected in a particular structural relationship they can define an object. (c) Different connections of the same geons can result in different objects as the figure shows geons 3 & 5 can give two different objects.

3. PRINCIPLES FOR DRAWING 3D DIAGRAMS

If as structural object recognition theories propose, the human visual system contains significant processing machinery designed to decompose the visual image into a set of generalized cone primitives, then we should be able to create more effective diagrams using these same primitives. Geons themselves represent a rich set of shape primitives that can be used to represent different classes of objects. Relationships between objects can be represented by the way geons are attached directly to one another, or elongated limb-like geons can be used as connecting structures. The secondary attributes of geons, namely color and surface texture can be used to represent secondary attributes of entities and relationships.

The following set of rules defines the "geon diagram".

- G1:** Major entities of a system should be represented using geons.
- G2:** The links between entities can be represented by the connections between geons. Thus the geon skeleton represents the topology of the information structure. In some cases certain relationships may also be represented by means of "limbs" consisting of elongated geons.
- G3:** Minor sub-components are represented as geon appendices – small geon components attached to larger geons.
- G4:** Attributes of entities and relationships are represented by geon color and texture and symbology mapped onto geons.

Although geon diagrams are 3D structures, the theory of Marr and Nishihara suggests that a good 2D layout will also be important in determining how readily geons are identified. The overall silhouette of the diagram will make it easier to identify the

program structure. In particular the joints between geons will be more readily identified if they are clearly identifiable in the X,Y plane. Thus we add the following two layout rules.

L1: The geon diagram should be laid out predominantly in the X,Y plane.

L2: Junctions between geons should be made clearly visible in the 2D plane, and should not overlap with other geon features.

4. EVALUATION OF GEON DIAGRAMS

In order to explore the effectiveness of geons in diagrams we first constructed a toolkit using the OpenGL 3D graphics standard. This toolkit makes it possible to construct geon diagrams according to the principles given previously. At present the geon toolkit is designed to be a research tool rather than a practical utility program and it lacks many of the features that would be essential in a usable system, the ability to attach text labels, for example. The toolkit:

- Allows building diagrams from a set of 24 geons,
- Is equipped with geons that can have surface properties such as color, texture, shading, and transparency,
- Provides for metric associations with the use of varying sizes and shapes of objects and their positioning in space,
- Provides for symbolic associations via surface properties,
- Formulates topological associations by the structural composition of geons.

The remainder of this paper describes three different studies that were designed to test the hypothesis that geon diagrams are easier to interpret and remember. In all the tasks the geon diagrams were compared to UML equivalents.

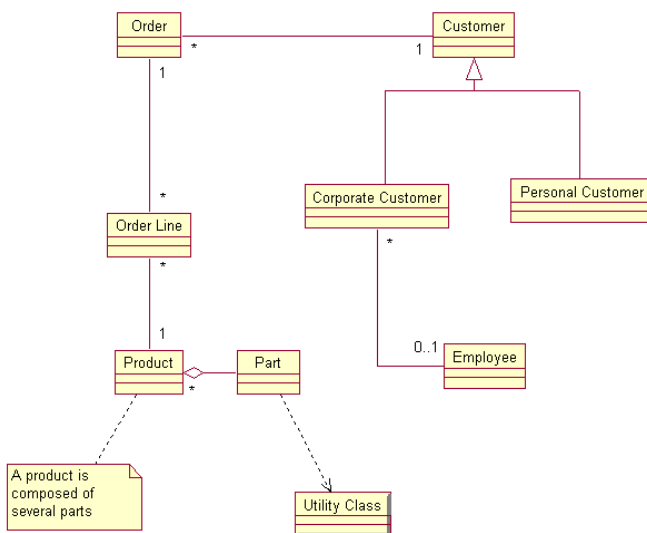


Figure 5. Sample UML diagram and geon equivalent.

The equivalent UML diagrams were drawn using a one-to-one mapping between geons and UML objects. The choice of UML was made as it is a rich diagramming notation that combines several diagramming techniques and it has become a de facto standard for many diagrams used in Software Engineering. For our experiments we chose to use UML class diagrams for object oriented programming. A sample UML diagram and its equivalent geon diagram are depicted in Figure 5.

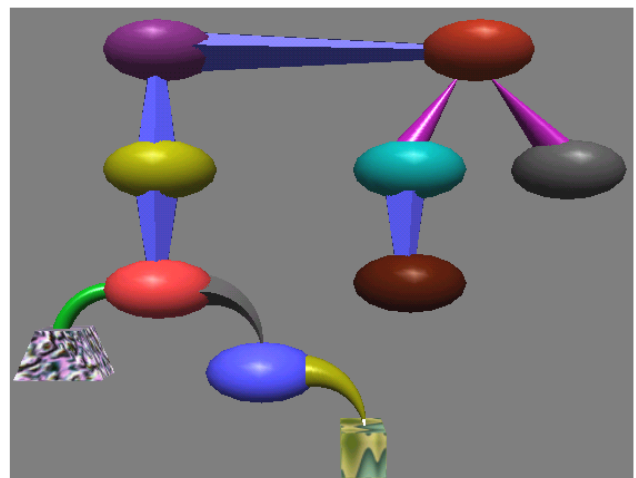
The ultimate goal of all three experiments is to determine whether geon diagrams are more easily interpreted and remembered. The first experiment is designed to determine the amount of time it takes a subject to recognize a sub-structure in UML versus geon diagrams. It also measures the accuracy of the identification. The second experiment investigates the accuracy of recalling diagrams. In the first two experiments the geon diagrams employ both color and texture to represent attributes. The third experiment eliminates all the factors of surface attributes that were included in experiments 1 and 2.

5. EXPERIMENT 1: SUB-STRUCTURE IDENTIFICATION WITH GEON DIAGRAMS VERSUS UML DIAGRAMS

The purpose of the first experiment was to determine the ease with which people can identify sub-structures in geon diagrams in comparison with equivalent UML diagrams. We measure the time and error rate for a subject to recognize a sub-structure of a diagram. We hypothesized that it should be possible to identify sub-structures faster and more accurately with geon diagrams.

5.1 Method for Experiment 1

Diagrams. Two sets of ten UML diagrams were drawn using Rational Rose™ and equivalent set of geon diagrams were



constructed using the geon toolkit. For each set a sub-structure was constructed; for the first sets the sub-structure contained 3 components (Figure 6) and for the second sets it contained 5 components. This sub-structure only was present in half the diagrams. When it was present the sub-structure had the same topology but was not a template match.

The UML diagrams did not depict any particular system but included most of the boxes and arcs used in UML class diagrams. The UML diagrams also used text to denote the names of classes and the type of the class (i.e. Parameterized Class A, or Utility Class B, etc.). The geon diagrams did not use any text labels and instead made use of color and texture to distinguish between the different types of classes. Diagrams were presented on a computer screen.

Procedure. On each trial the subject was first shown the sub-structure for 15 seconds and was then given to run a set of 3 practice trials. The program selected a diagram randomly from the set of 10 diagrams and presented it to the user. They could press the 'Y' key if the sub-structure was present or else press the 'N' key. The response time of the user was captured along with the accuracy of the response.

The order in which the sets were presented to subjects was randomly selected as follows where G denotes a set of geon diagrams and U a set of UML diagrams: {G1,U1,U2,G2}, {U1,G1,G2,U2}, {U1,G1,U2,G2}, {G1,U1,G2,U2}.

The 15 subjects were all computer science students.

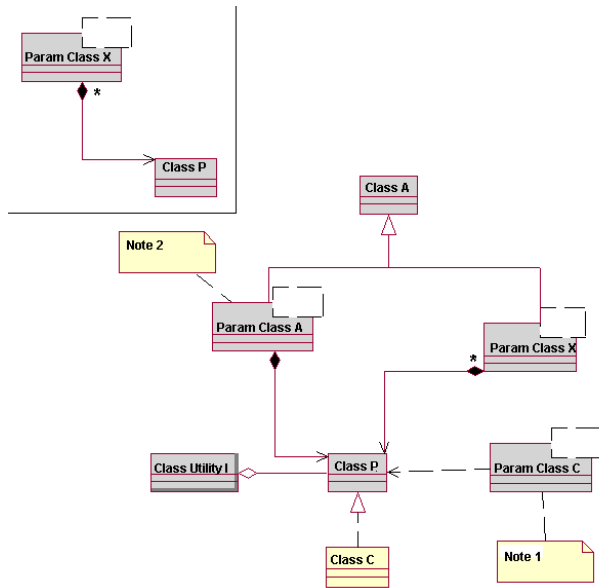


Figure 6. a) A simple UML sub-structure that users were to identify in Experiment 1 is depicted in the upper left-hand corner and a UML diagram containing the sub-structure. b) Geon sub-structure and diagram containing the sub-structure. The sub-structure was placed in differing orientations than originally shown to the subject.

5.2 Results of Experiment 1

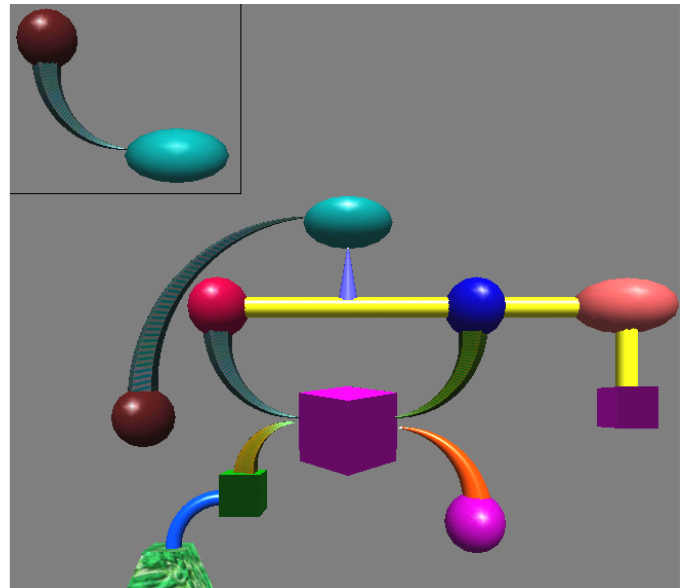
Results are summarized in Table 1. These show that substructures were identified both faster and more accurately with the geon diagrams. From the 15 subjects, 11 subjects correctly identified the sub-structure in more geon than UML diagrams, 1 subject identified the sub-structure equally often with the geon diagrams as with the UML diagrams, and the remaining 3 were more accurate with the UML diagrams. A sign test shows this to be significant ($p < 0.05$).

On average the subjects took 4.3 seconds to identify (correctly or incorrectly) the presence of the sub-structure in the geon diagram and 7.1 seconds for the UML diagrams. Of the set of 15 subjects, 13 identified the geon sub-structure faster than the UML sub-structure. A t-test shows this difference to be highly significant ($p < 0.005$).

These results support the hypothesis that geon diagrams are easier and faster to interpret than 2D UML diagrams.

	<u>Geon Diagram</u>	<u>UML Diagram</u>
Identification Time (sec)	4.3	7.1
Error Rate	13.33%	26.33%

Table 1. Summary of Results of Experiment 1.



6. EXPERIMENT 2: RECALL OF GEON DIAGRAMS VERSUS NON-GEON DIAGRAMS

The purpose of the second experiment was to determine whether geon diagrams can be remembered more easily than UML diagrams. Under certain circumstances human memory for image memory is remarkably good although not for abstract images [5]. We hypothesized that since geon diagrams more resemble real-world structures they might be more readily recalled after a brief exposure than comparable UML diagrams.

6.1 Method for Experiment 2

To determine whether a user can easily recall a recognized set of geon diagrams, this experiment was formulated to compare the accuracy of recalling geon diagrams versus equivalent UML diagrams.

Diagrams. A set of 14 UML diagrams was developed using Rational Rose™ UML designer. Using our mapping convention of geons to UML entities, an equivalent set of 14 geon diagrams was produced using the geon toolkit. Both sets of diagrams were printed in color on 8.5 by 11" transparencies.

Procedure. The experiment was conducted using two sets of students in senior level computer science courses. The experiment was performed in a classroom setting to students who had never performed the experiment earlier (if a student had performed a similar experiment in a previous lecture, they were asked to indicate that on the handout and the result was later discarded).

At the beginning of the lecture the first set of students were shown half (seven) of the set of geon diagrams in random order for 15 seconds per diagram. After presenting the first half of geon diagrams they were then presented with seven UML diagrams at intervals of 15 seconds. At the end of the lecture, or fifty minutes later, the students were then shown the full set of 14 geon diagrams and then 14 UML diagrams. Each diagram was shown for 10 seconds and the subject would indicate on a printed sheet whether that diagram had been part of the initial set. To counterbalance the first set of results, the same procedure was applied to the second set of students with the UML diagrams being presented first and the geon diagrams second.

There were 18 students that participated with geon diagrams first and 17 students that participated with UML diagrams first giving a total of 35 subjects. All the students were familiar with UML notation.

6.2 Results of Experiment 2

Subjects made less than half the errors in recalling geon diagrams than they did for the equivalent UML diagrams; 18% error rate for the geon diagrams vs. 39% for the UML diagrams. This difference is especially striking considering that chance performance is 50%.

From the 35 subjects, 26 recalled correctly more geon than UML diagrams while 5 recalled the same number of geon diagrams as UML diagrams and 4 subject recalled more UML diagrams. A sign test showed this difference to be highly significant ($p < 0.005$).

These results support the hypothesis that geon diagrams are easier to remember.

7. EXPERIMENT 3: RECALL OF GEON DIAGRAMS VERSUS NON-GEON DIAGRAMS WITHOUT SURFACE ATTRIBUTES

The geon diagrams used in Experiments 1 and 2 used geons that were distinguished using color and surface texture as well as 3D geon shape. It might therefore be the case that color and texture were more important than the use of 3D geons in making these diagrams more effective. The third experiment was designed to determine whether geon diagrams without surface attributes would still be better than the recall of UML diagrams.

7.1 Method for Experiment 3

The procedure was the same as for Experiment 2. The first batch of students were first shown the UML diagrams and the second batch of students were shown the geon diagrams first. There were a total of 25 students in the first batch and 17 students in the second. The data from those students who had performed a similar experiment before were discarded as well as three randomly chosen result sets to bring the total number of subjects to 35 as in experiment 2.

7.2 Results of Experiment 3

Subjects on average had an error rate of 22.5% for recalling the geon diagrams and 42% for recalling the equivalent 2D UML diagrams; or almost half the error rates were observed with the geon diagrams. From a total of 35 subjects, 25 recalled more geon diagrams, 2 recalled the same number of geon diagrams as UML diagrams, and 8 subjects recalled more UML diagrams. This difference is significant ($p < 0.01$).

This experiment, supports the hypothesis that remembering geon diagrams is easier than remembering UML diagrams even when the geon diagrams are not presented with surface attributes.

8. CONCLUSION

We have argued that 3D diagrams using geon primitives may provide a better match to high-level processes that occurs in human object recognition and because of this they should be easier to interpret and remember. Our experiments generally support this hypothesis. Experiment 1 shows that users are quicker in identifying the geon sub-structure in comparison with an identically structured UML diagram. Experiment 2 shows that that geon diagrams are much easier to remember than UML

diagrams, however, the use of color and textures might have accounted for this advantage as much as the use of 3D primitives. For experiment 3 the geon diagrams were not equipped with any surface attributes and in addition the UML diagrams did not contain any labels to distinguish the objects. The only relevant factor of this experiment was the structural impact the diagrams presented. Results from this experiment also show that users are able to memorize better the structure of the diagrams constructed using geon primitives.

Nevertheless, although our results are encouraging, and certainly show that UML diagrams can be improved, we cannot claim to have demonstrated in any rigorous way that the use of 3D primitives is the key factor. It might be that the outline shapes of the diagrams we constructed are more readily distinguished and remembered than the shapes used in UML diagrams. We chose UML diagrams because they are the de facto standard for describing information systems. However, we are planning experiments to specifically test the importance of using 3D primitives.

There are inevitably tradeoffs inherent in creating geon diagrams. The complexity of what can be represented using these kinds of primitives may be less than what is possible using more cryptic line and box diagramming techniques. Another issue that must be addressed when using geon diagrams is how to effectively label the relationships and nodes. For example, it may be difficult to show text as clearly on a 3D shaded object. If the object is textured this is especially likely to interfere with the readability unless the texture is subtle.

There is much more research to be done.

- An investigation of the various types of relationships that can effectively be displayed using geon objects. As a start this may be based on Biederman's set of interconnection rules.
- An investigation of the types of surface attributes that can convey information without clutter

To reiterate our main finding, our results support the use 3D structured primitives for drawing diagrams. These diagrams are easier to interpret and remember than commonly used UML diagrams.

9. REFERENCES

- [1] Bertin, J. (1981) Graphics and Graphic Information Processing, Walter de Gruyter & Co. Berlin.
- [2] Biederman, I. (1987) Recognition-by-Components: A Theory of Human Image Understanding, *Psychological Review*, Vol. 94, No. 2, 115-147.
- [3] Biederman, I. and Gerhardstein, P.C. (1993) Recognizing depth-rotated objects: Evidence and conditions for three dimensional viewpoint invariance, *Journal of Experimental Psychology: Human Perception and Performance*, 19, 1162-1182.
- [4] Biederman, I. and Gerhardstein, P. C. (1995) Viewpoint-Dependent Mechanisms in Visual Object Recognition: Reply to Tarr and Bulthoff (1995), *Journal of Experimental Psychology*, 21:6, 1995, pp. 1506-1514.
- [5] Bower, G.H., Carlin, M.B., and Duek, A., (1975) Comprehension and memory for pictures. *Memory and Cognition*, 3(2) 216-220.
- [6] Fowler, M. (1996) *UML Distilled: Applying the Standard Object Modeling Language*, Addison-Wesley.
- [7] Halverston, J. (1992) The first pictures, perceptual foundations of Paleolithic art, *Perception*, 21,389-404.
- [8] Harel, D., (1995) On Visual Formalisms, In J. Glasgow, N.H. Narayanan, and B. Chandrasekaran, *Diagrammatic Reasoning: Cognitive and Computational Perspectives*, MIT Press. 235-272.
- [9] Hummel and Biederman (1992) Dynamic binding in a neural network for shape recognition. *Psychological Review*, 99, 480-517.
- [10] Koffka, K. (1935) *Principles of Gestalt Psychology*, New York, Harcourt.
- [11] Kosslyn, S.M. (1994) *Elements of Graph Design*. W.H. Freeman and Company, NY.
- [12] Marr, D. (1982) *Vision: A computational investigation into the human representation and processing of visual information*. San Fransisco, CA: Freeman.
- [13] Marr, D., and Nishihara, H.K. (1978) Representation and recognition of the spatial organization of three-dimensional shapes. *Proceedings of the Royal Society of London, B*, 200, 269-294.
- [14] Pressman, R.S. (1992) *Software Engineering: A Practitioner's Approach*, McGraw-Hill.
- [15] Scaife, M. and Rogers, Y. (1996) External cognition: how do graphical representation work? *International Journal of Human-Computer Studies*, 45, 185-213.
- [16] Ullman, S. (1989) Aligning pictorial descriptions: An approach to object recognition, *Cognition*, 32, 193-254.
- [17] Ware, C. (2000) *Information Visualization: Perception for Design*, Morgan Kaufman.
- [18] Zhang, J. (1997) The nature of external representation in problem solving, *Cognitive Science*, 21(2), 179-217.