NETWORK REPRESENTATION AND ROTATION OF LETTERS (1)

J. A. Levin

Department of Psychology

UCSD, La Jolla, CA 92037

September 12, 1973

Introduction

This paper concerns an area that now is the subject of controversy: Is the representation of "sensory" information in long-term memory "propositional" (Pylyshyn, 1973; Palmer, 1973) or is the representation "analog" (Sloman, 1971)?
My own feeling about this is that the first five minutes of such a discussion are valuable; the next N years of argument are fruitless. In this paper, I present a specific (and therefore probably incorrect) representation of the letters of the alphabet and a particular process model for mental rotation. My hope is that these specific models will spur the development of more concrete theories of the representation and manipulation of "sensory" information in long-term memory.

The Representation of Letters

Early approaches to the representation of visual phenomenon, especially letters, have been of two varieties: 1) template systems, from the "analog" tradition, and 2) feature models, from the propositional side. Both of these systems have their shortcomings (see Palmer, 1973).

Recently, network models have been advanced as a system for representing sensory phenomenon. Pat Winston, in his thesis (Winston, 1970), presented a system for representing, recognizing, and learning spatial configurations, such as arches. I will apply the same general approach in this paper, with some twists.

Early feature models would characterize an A as:

1 horizontal line, 1 left oblique line, 1 right oblique line. Clearly, this is not a sufficient representation, as it doesn't rule out configurations that are clearly not A's, as in Figure 1.

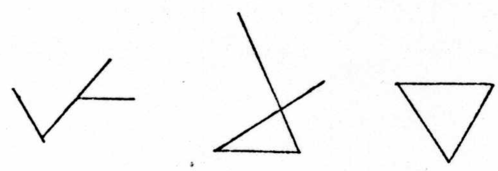


Figure 1: Non A's accepted by a feature model as A

We need to augment this representation with the spatial relationships between the features. What are the crucial relationships in an A? Well, the ends of the diagonals touch, and each end of the horizontal line intersects the middle of each diagonal. Let's develop a network representation of such an A (see Footnote 2 about the notational conventions used). First, there are the 3 lines, as shown in Figure 2a. Next, there are the parts of the lines that touch, shown in Figure 2b. Finally, let's add the relation "NEAR" between those parts of the line that are in close physical proximity, as shown in Figure 2c.

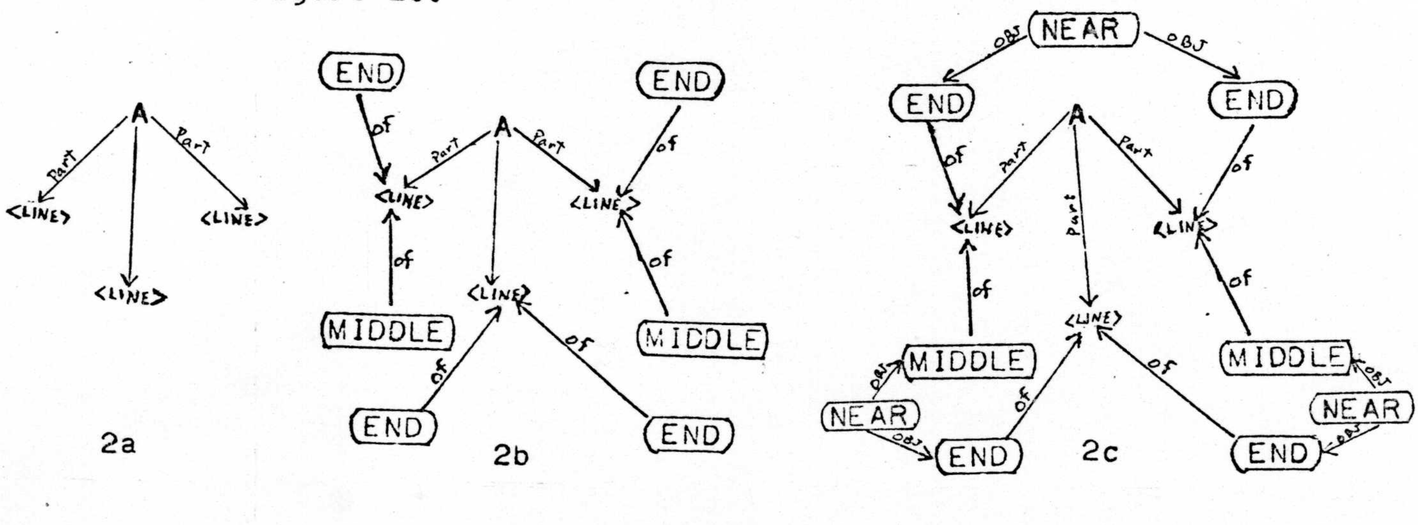


Figure 2: Network representation of an A

Examples of some spatial configurations allowed by this definition are shown in Figure 3. These are all A's, but in all different orientations.

Figure 3: Different configurations specified by the definition

The definition we have just developed is an "orientation-free" representation of an A.

(To test the adequecy of any representaion claimed, here or elsewhere, you should try to find some configuration that meets the definition's constraints, but which wouldn't be considered in the category. For example, does the above definition (in Figure 2c) rule out all non A's?

Another side point: the definition above is clearly not a complete description of A, since it doesn't describe script A's or small A's. This problem is an issue I will ignore for now.)

People can easily distinguish between an upside-down A and a right-side up one. Can we represent this orientation information using the definition we have developed?

First of all, how is orientation specified? For an A, the intersection of the two side lines must be at the top and the intersections of the sides and the crossbar must be at the sides.

Before we can tackle the problem of representing orientation in a network, we must take a side trip into the issue of how to represent orientation. It is in the representation of "continuous" dimensions like orientation, that the propositional vs. analog controversy gets its hotest. First of all, do we really need a continuum - that is, do we need to use an infinite number of values to represent orientation? Clearly not - the human perceptual system just doesn't have infinite sensitivity. At the lowest perceptual level, there is a finite change in orientation that has to be made for a human to detect any change. Smaller changes are just not perceived. So, in fact, we need only represent discrete orientation categories, into which an infinite number of physical orientations map. The existence of these orientation equivalence classes doesn't imply that orientation can't be perceived as "continuous". As long as the size of these classes is smaller than the "grain" of perception, rotation through discrete orientation classes will be perceived as continuous. A good example is the perception of a movie of a rotating object. In this case, even the external stimuli are "discrete" - only discrete orientations of the object are presented. But we still perceive continuous motion.

So let us assume that there is some set of equivalence classes of orientations. Certainly, a psychologically valid set of these have to be established empirically. (There's even some evidence that these classes are not of equal size - human have finer discrimination along horizontal and vertical than on diagonals). But let us adopt the following compass point set, mainly for mnemonic value, described in Figure 4.

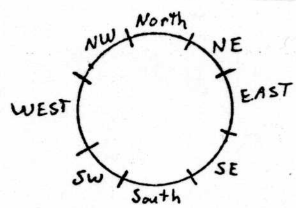


Figure 4: Hypothetical orientation equivalence classes

By North, we really mean "up", East is "right", etc.

An Upright A

Now, back to A. How do we augment our orientation-free representation to represent a particularly oriented one? The specification of the relative position of the various parts has to be in relation to some particular position. So we need to add a reference point for each definition. Then, if we add the necessary relative position predicates between the reference point and the orientation sensitive parts of the network - presto, we have an upright A.

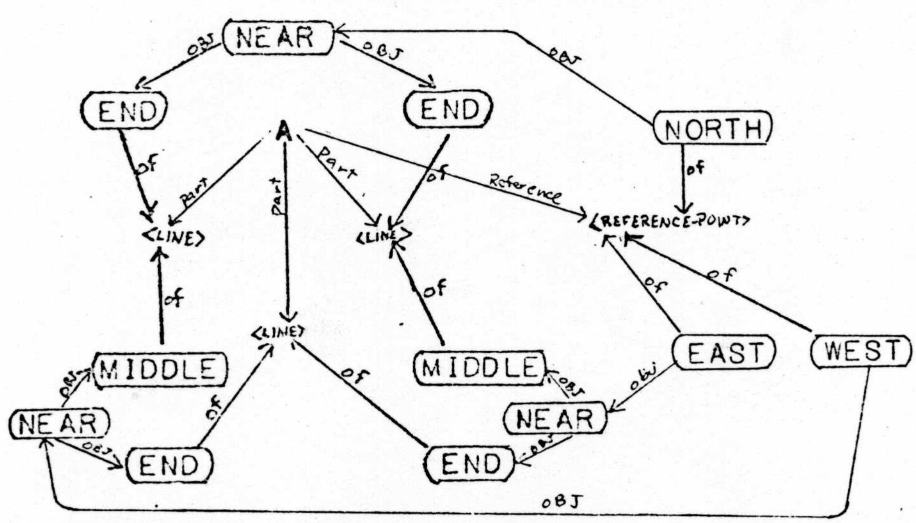


Figure 5: Representation of an upright A.

Just for fun, here is the representation of a tilted A.

See if you can derive the orientation from the representation!

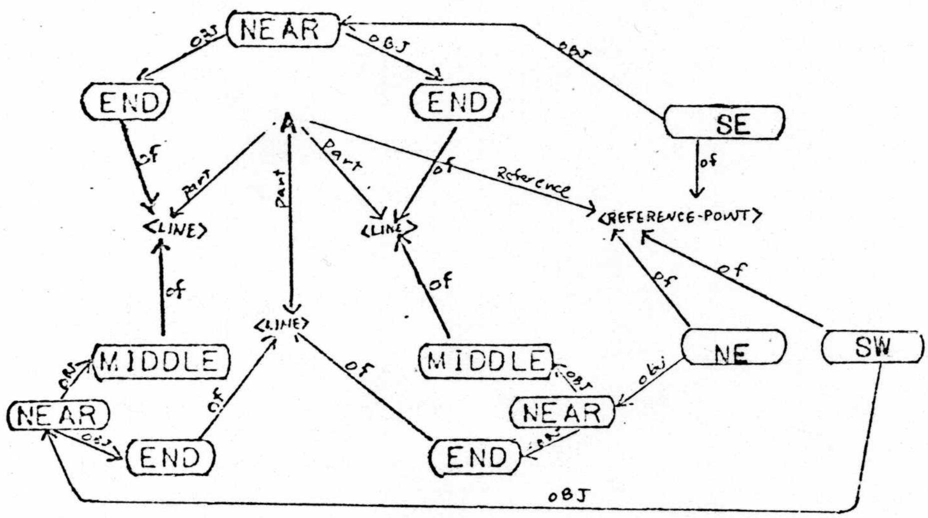


Figure 6: Representation of a tilted A.

Note that the reference points for these two representations are located in the center of the crossbar of the A.

You can test the naturalness of my representations by picking any letter, say an N, and drawing your representation below. You can then check it against mine in the Appendix.

Figure 7: Reader's Representation of an N.

See the Appendix for the orientation-free and orientation sensitive (upright) representations of all of the letters.

Let's summarize the representational system used for letters. The basic building blocks are lines and curves. A line is a general concept that includes both stright line segments and curves. By a curve, I mean a simple curved line segment (one with no inflection points - one who's radius of curvature never crosses the curve. .We have operators which specify parts of these basic elements: END and MIDDLE. Lastly, we have a relation predicate NEAR which specifies physical proximity between its arguments.

With these components, we are able to represent with some degree of success orientation-free definitions of the letters. To specify particular orientations, we have to add a reference point and a set of orientation predicates which relate the orientation sensitive subparts of the representation to the reference point.

(To represent more complex figures, we can use these elements to define higher level units. See Footnote 3 for an example of this.)

O.K., so what do we do with all these networks? Well, several things. First, if these representations really describe the class of configurations of printed capital letters, then we should be able to use them to recognize these letters. In this view, the perceptual processes construct a network representing the external stimulus and a network matching process would identify the letter.

Secondly, we can do various mental manipulations of letters using these representations. For example, we could rotate them. Let us assume that the equivalence classes of orientations (discussed above) are stored in a semantic network with the following relations between successive classes:

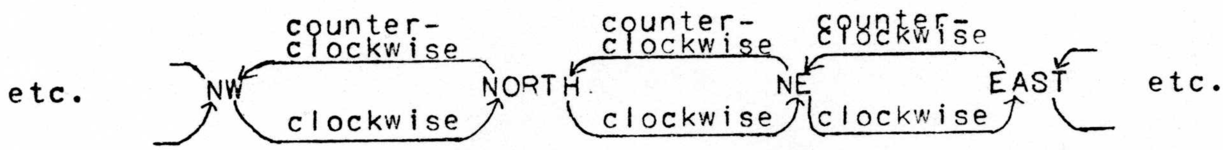


Figure 8: Network representation of orientation

Now, say we wanted to rotate an upright A one step clockwise. That is, we want to change Figure 5 into Figure 9 below.

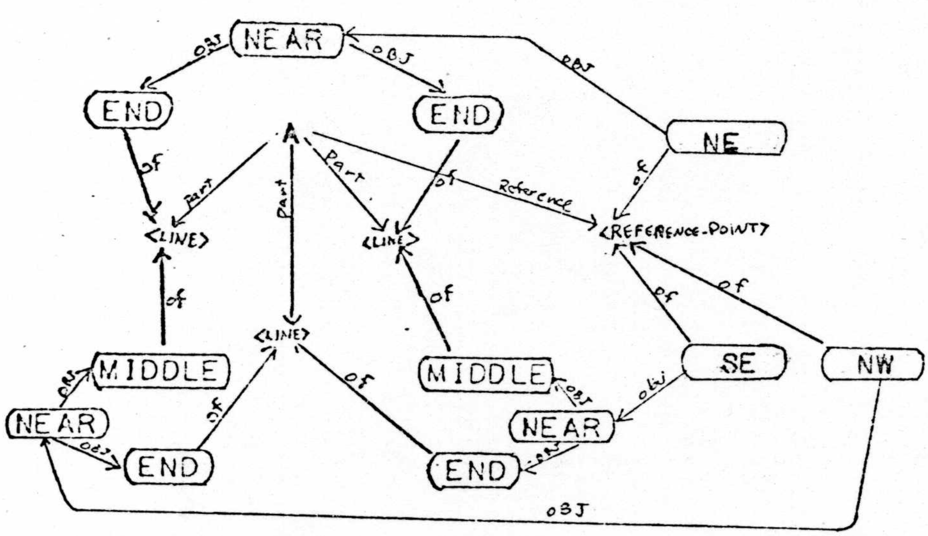


Figure 9: Representation of an A.

What did we have to do to Figure 5 to get Figure 9? Basically, we replaced each orientation predicate relation with the next orientation relation in the clockwise direction. We can find this next predicate by following the CLOCKWISE relation from the type node of the specified predicate. For example, NORTH is replaced by NE (north-east), which is the node we get to from NORTH by CLOCKWISE, as shown in Figure 8. Here is a SOL definition (see footnote 4) for the procedure ROTATE1, which will rotate a specified letter one step in the specified direction:

```
> Define Rotate1 as action.
THE DEFINITION FRAME FOR ROTATE1 IS:
* Rotate1 object direction.
THE DEFINITION IS:
* Call the first node from object via reference reference-point.
* Change each node from reference-point via OF-I
*       from "act" firstnode from it via "act"
*       to "act" firstnode from that node via direction.
* ##
```

Now we can easily write the general procedure Rotate.

```
> Define Rotate as action.
THE DEFINITION FRAME FOR ROTATE IS:
* Rotate object number-of-steps direction.
THE DEFINITION IS:
* Set a  counter to 1.
* Rotate1 object direction and increment the counter
*      until the counter is greater than number-of-steps.
* ##
```

Now let's look at the mental rotation task used by Shepard and his associates (Shepard & Metzler, 1971; Cooper and Shepard, 1972). In these tasks, subject were required to mentally rotate images to determine whether they matched perceptually presented objects. The interesting result of all these studies is that the more the mental image has to be rotated, the longer it takes. In fact, the relation between time to rotate and amount of rotation is linear. This relation is shown in Figure 10.

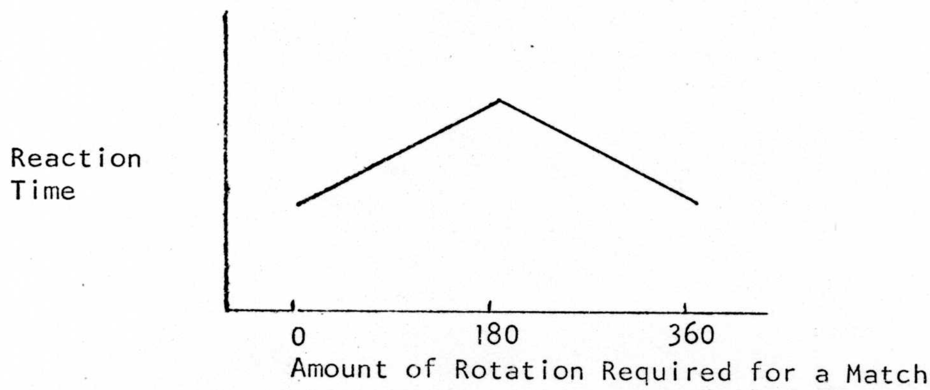


Figure 10: Data from mental rotation experiments

Now lets's see if we can simulate the processes the subjects use in these experimental tasks. Let us define an overall procedure COMPARE.

> Define Compare as action.

THE DEFINITION FRAME OF COMPARE IS:

* Compare mental-image to perceptual-image.

THE DEFINITION IS:

* Rotate1 mental-image clockwise until a match between mental-image
*   and perceptual-image occurs.
* If the match is same as identical, respond "yes",
*   otherwise respond "no". ##

Let's assume that MATCH is some relatively simple network matching procedure. All other words in this definition are either defined previously in this paper (ROTATE1) or are standard words in SOL.

Before we go on, let's take a little digression to note a flaw in this definition. You'll note that it "blindly" rotates the image clockwise until a match is found. If people really did this, the data would look like that in Figure 11.

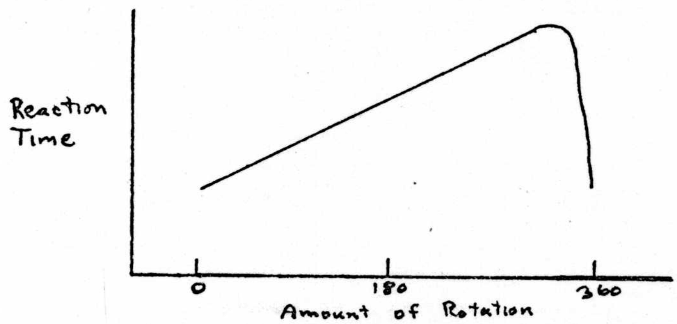


Figure 11: Fallacious predicted data

Another alternative, to randomly choose which way to rotate, would produce the results shown in Figure 12.

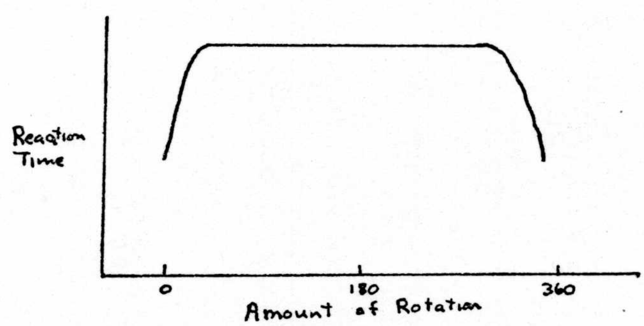


Figure 12: More fallacious predicted data

Somehow, people "know" which way to start rotating the object. I'll leave this as a problem for the reader (and for myself). Note that this problem exists for any model of the rotation process, whether analog, propositional or in-between.

Anyway, we can develop an explicit prediction for the reaction times in these rotation tasks from these process models:

S = number of steps needed to rotate the image into correspondence with the perceptual input

M = time to execute the MATCH procedure, to check the result, and to execute the ROTATE1 procedure

C = constant time to perceive the problem and to make the response (this is the garbage category)

then, RT = S * M + C i.e. the straight line data of Shepard et al.

Now, let's look at a nice variation of this paradigm, developed by Lynn Cooper (Cooper & Shepard, 1972). In this version, subjects are told to mentally rotate a particular letter clockwise in time to a rhythm (a tape recorder plays a monolog like this "...up, tip, tip, down, tip, tip, up, ..."). At some time unknown to the subjects, a real version of the same letter is presented in some orientation, and the subjects determine whether the presented letter is the same of the one they're rotating or its mirror image.

Let's look at a possible model of what subjects do in this task.

> Define "Do-Lynn's-experiment" as action.

THE DEFINITION FRAME FOR DO-LYNN'S-EXPERIMENT IS:

* Do-Lynn's-experiment with letter.

THE DEFINITION IS:

* Rotate1 letter clockwise until a perceptual-input occurs,

*   then compare letter to the perceptual-input.   ##

Again, the prediction of this model is the straight line function found in this experiment.

The rotation model presented here has another interesting prediction, derived from the definition of ROTATE1 (given above). From the definition, we can predict that it will take linearly longer to rotate a figure with more orientationally sensitive points. This prediction agrees with the intuitive notion that it is harder to rotate some objects. In fact, this is experimentally verified by Shepard et al.: to rotate complex block figures takes about 20 msec./degree (Shepard and Metzler, 1971); to rotate simplier figures (letters) takes about 2 msec./degree (Cooper and Shepard, 1972). There are two points to note concerning this prediction.

1) This difference in rate depends only on the number of orientation sensitive nodes, <u>not</u> on the overall complexity of the representation. Much of the orientation-free representation isn't involved in the rotation process (it comes along for a free ride).

2) This prediction is evidence against extreme analog models. Physically, it's no harder to rotate a visually complex object than a visually simple one (of equal mass). Considerable abstraction must occur before "visual complexity" can even enter into a rotation process.

RANDOM IDEAS, AN APOLOGY, AND A QUICK SUMMARY

The representations of letters presented here are open to many other experimental tests. To pick a random sample:

1) The representations give a measure of visual complexity, which can be tested (note the distinction between visual complexity and orientational complexity mentioned previously).
2) They provide a measure of the similarity of any set of objects. A direct test of these representations would be to see how well they predict the table of letter recognition errors.
3) They provide a database for a model of recognition. In particular, the letter representations provide an alternative set of basic elements for the various models of letter recognition.

The ideas here are not completely worked out, and certainly not experimentally verified. My hope (and justification for writing this up) is that this paper will spur the development of concrete ideas, specific models, and informative experiments in this area, so that we can go beyond futile black or white arguments of propositional vs. analog representations.

In this paper, I have presented a representational system for specifying orientation-free representations of the printed capital letter. In addition, a scheme for representing orientation was developed. The complete set of these representations is in the Appendix. Using this representation system, we were able to easily write models of the mental rotation processes. These models satisfied the gross constraints of the data, and in addition, provided an explanation for the differences between objects in difficulty of rotation.

NOTES:

(1) This research is supported by research grant GB 32235X from the National Science Foundation. Many of the concepts discussed in this paper have come from the activities of the LNR Research Group. This paper has been especially influenced by Steve Palmer, both in personal discussion and by his paper summarizing these issues (Palmer, 1973). The particular representational system and the model of rotation presented here were developed at the Workshops on Human Information Processing, at Carnegie-Mellon University, June-July, 1973.

(2) Here is a short summary of the notation used in these graphs:

relation R between nodes A and B is denoted like this:

$$A \xrightarrow{R} B$$

The converse relation from B to A is also symbolized by the same arrow. If it isn't assigned a name, it has the default name of "R-I". Relations used in the representations presented here include PART, symbolizing that B is a subpart of A;

REFERENCE, pointing to the reference point of a definition;

OF, marking one argument of the operators;

OBJ or O, marking unordered arguments of predicates.

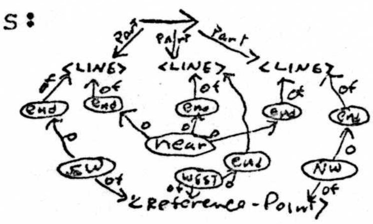
Concept token nodes for the type concept A are noted as <A>.

This means that the <A> node stands for a particular instance of the general concept A.

Predicate token nodes for the type A are noted as A

(3) Larger scale units can be defined from the units used here. For example, to analyze complex three-dimensional units, you might want to define the units used by Guzman in his scene analysis program (Guzman, 1969):

Unit	Defined as
L	Like "L" in Appendix.
T	Like "T" in Appendix.
X	Like "X" in Appendix.
K	Like "K" in Appendix
Fork	Like first "Y" in Appendix.
Arrow	Like this:



(4) SOL is a language developed by our group for ease in writing models of cognitive processes. See "The Memod Manual" if you want to understand the nitty-gritty details of these definitions. These definitions consist of 1) the specification of the name and syntactic class, 2) the specification of the argument frame, 3) the body of the definition, specifying uhe actions to be performed, 4) the end of definition marker ##. User input is in lower case; computer responses in UPPER.

REFERENCES

Cooper, L.A. and Shepard, R.N., Chronometric Studies of the Rotation of Mental Images. In W.G. Chase (Ed.), *Visual Information Processing*, New York: Academic Press, 1973.

Guzman, A., Decomposition of a visual scene into three-dimensional bodies. In A. Grasselli(ed.), *Automatic interpretation and classification of images*. New York: Academic Press, 1969

LNR Research Group, MEMOD: A Model for the Representation of Language and Semantic Structures in Human Long-term Memory. Center for Human Information Processing, UCSD, La Jolla: June 1973

Palmer, S.E. The Problem of Visual Representation. Manuscript, U.C.S.D., La Jolla: January 1973.

Pylyshyn, Z.W. What the Mind's Eye Tells the Mind's Brain: A Critique of Mental Imagery. *Psychological Bulletin*, 1973, *80*(1), 1-24

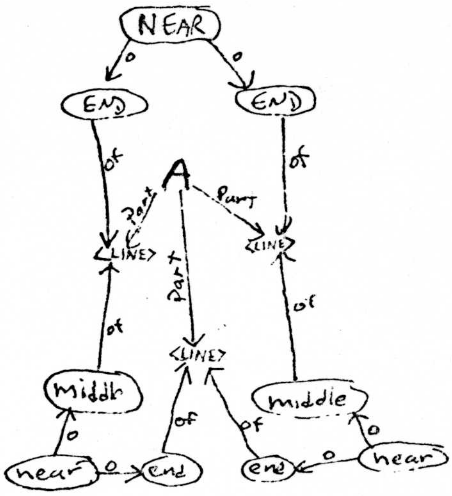
Shepard, R.N. and Metzler, J. Mental rotation of three-dimensional objects. *Science*, 1971, *171*, 701-703.

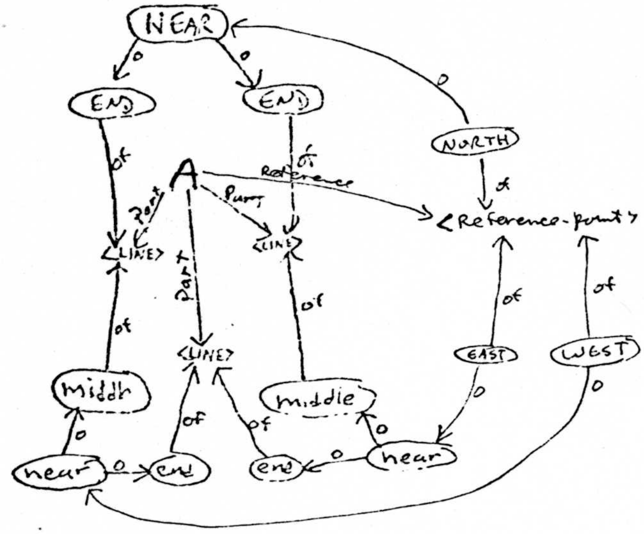
Sloman, A. Interactions Between Philosophy and Artificial Intelligence: The Role of Intuition and Non-Logical Reasoning in Intelligence. *Artificial Intelligence*, *2*, 209-225

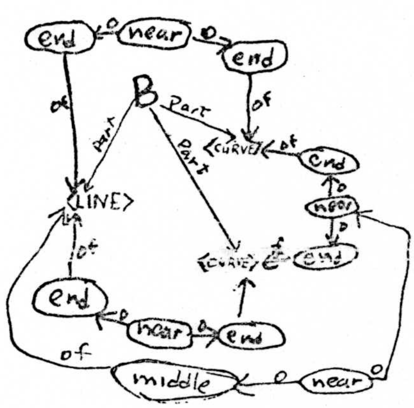
Winston, P.H. Learning structural descriptions from examples. M.I.T. Project MAC TR-76, 1970

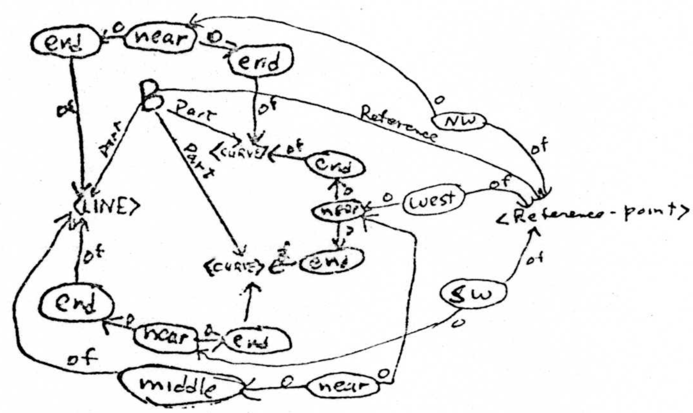
APPENDIX: ORIENTATION--FREE AND ORIENTATION SENSITIVE (upright)

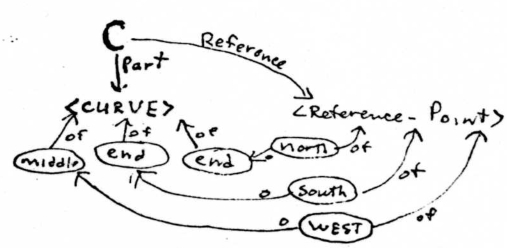
REPRESENTATIONS OF LETTERS

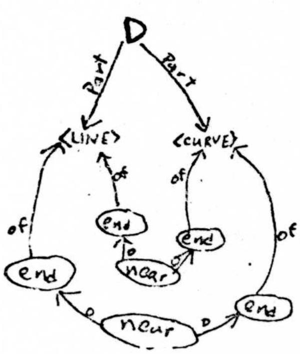


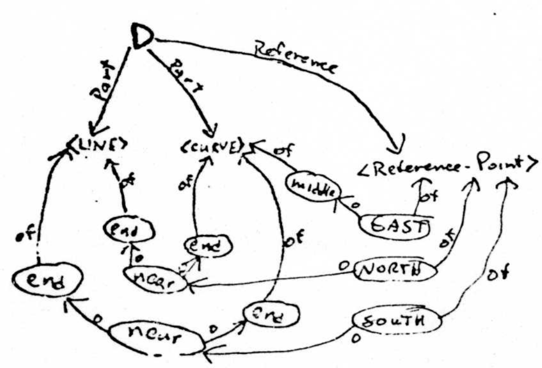




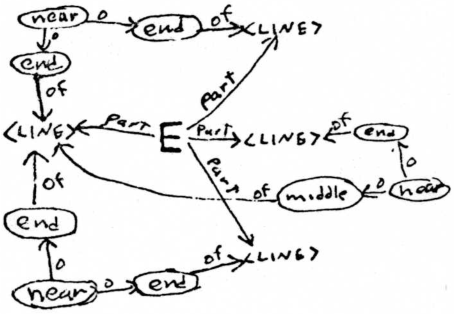


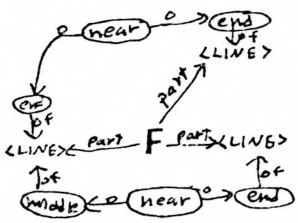


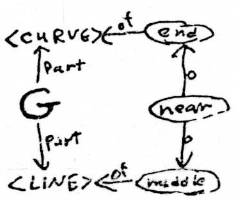


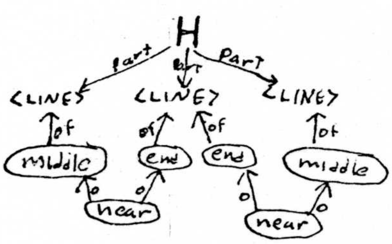


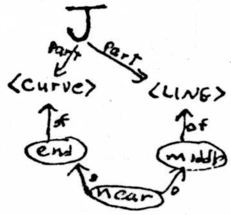
Orientation-free



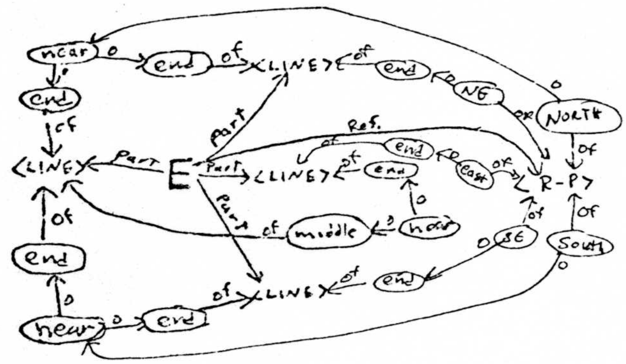


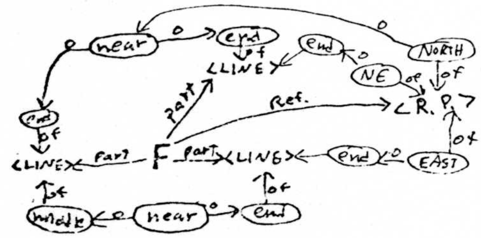


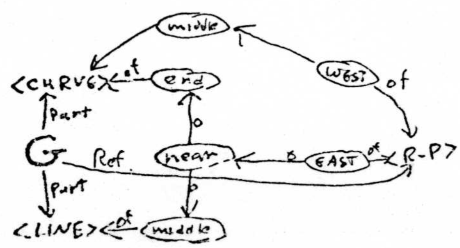


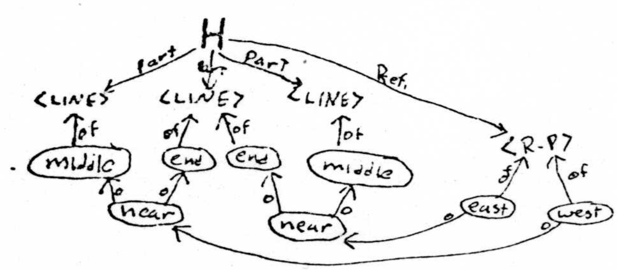


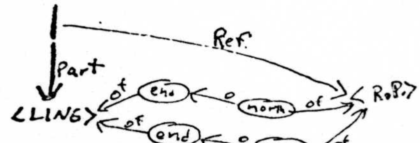
Orientation-sensitive

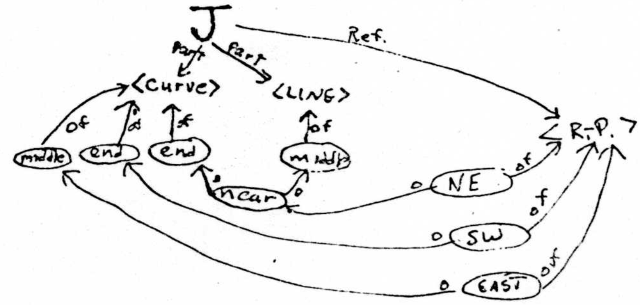


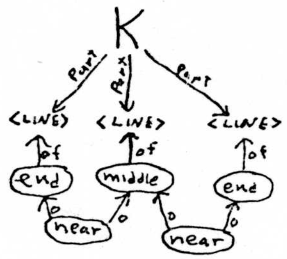


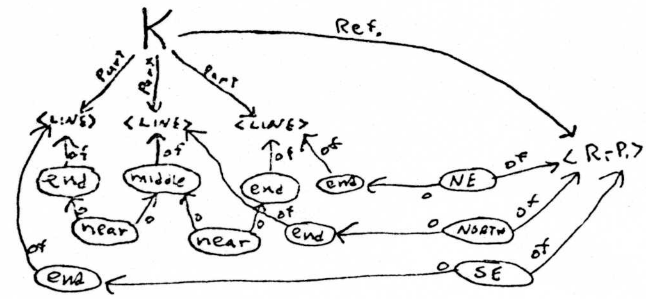


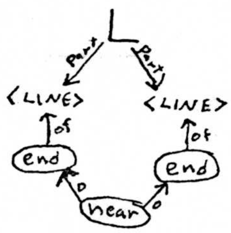


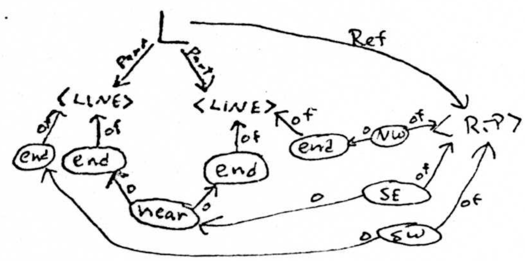


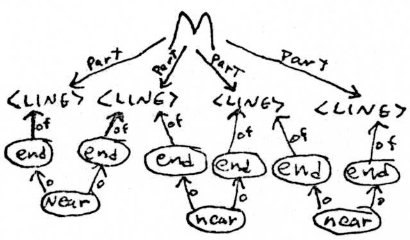


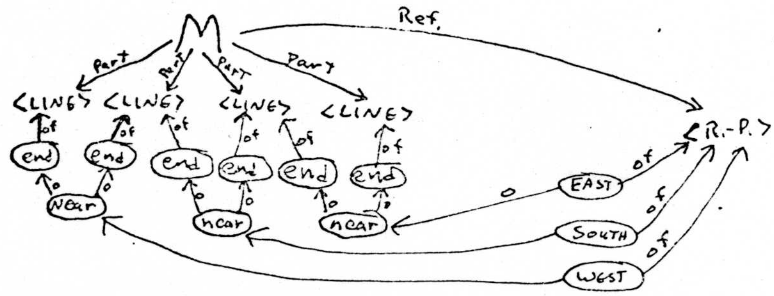

K
part
part
part
<LINE>
<LINE>
<LINE>
of
end
middle
end
near
near

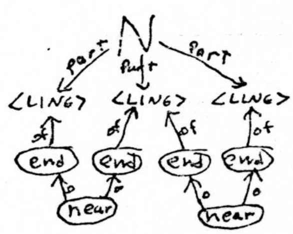

K
Ref.
part
part
part
<LINE>
<LINE>
<LINE>
<R.-P.>
end
middle
end
end
end
end
near
near
NE
NORTH
SE

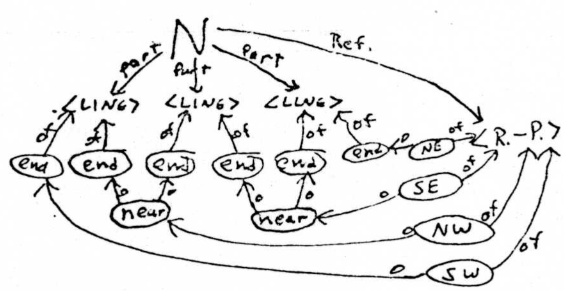

L
part
part
<LINE>
<LINE>
end
end
near

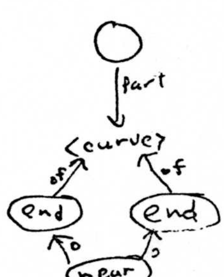

L
Ref
part
part
<LINE>
<LINE>
<R.-P.>
end
end
end
end
NW
SE
SW
near

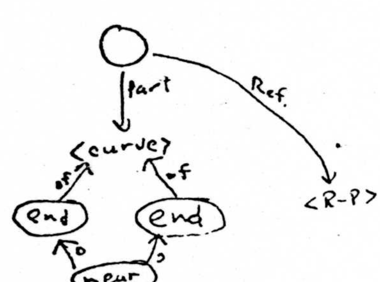

M
part
part
part
part
<LINE>
<LINE>
<LINE>
<LINE>
end
near
near
near

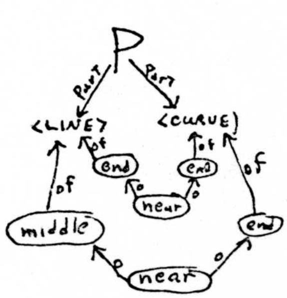

M
Ref.
part
part
part
part
<LINE>
<LINE>
<LINE>
<LINE>
<R.-P.>
end
near
near
near
EAST
SOUTH
WEST

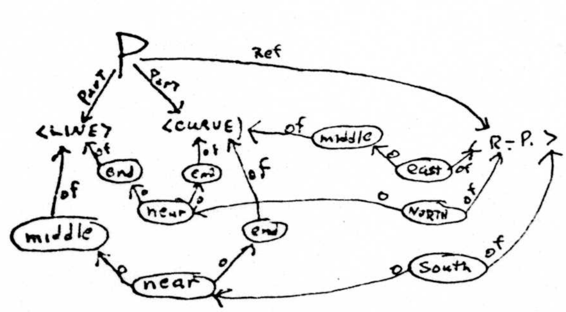

N
part
part
part
<LINE>
<LINE>
<LINE>
end
near
near


N
Ref.
part
part
part
<LINE>
<LINE>
<LINE>
<R.-P.>
end
NE
SE
NW
SW
near
near


O
part
<curve>
end
end
near

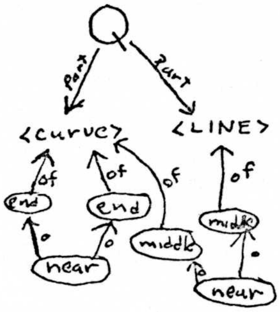

O
Ref.
part
<curve>
<R-P>
end
end
near

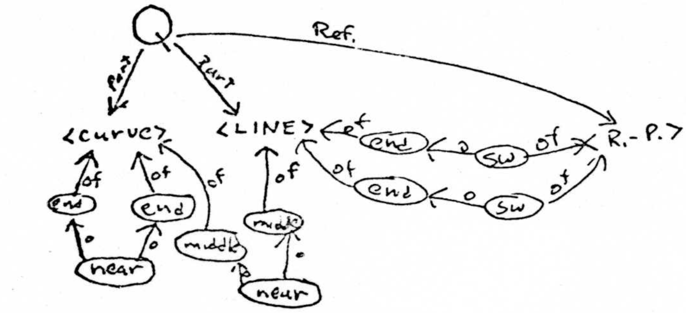

P
part
part
<LINE>
<CURVE>
end
end
near
middle
end
near

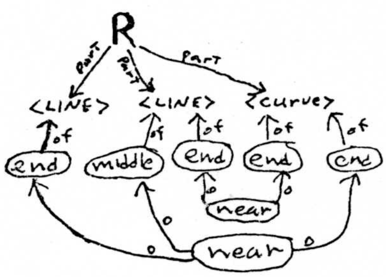

P
Ref
part
part
<LINE>
<CURVE>
<R-P.>
middle
east
NORTH
South
end
end
near
middle
end
near

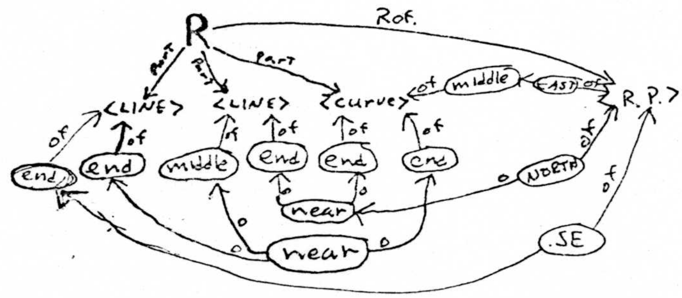
Orientation-free

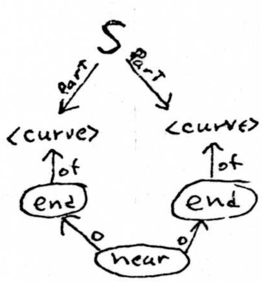
Orientation-sensitive

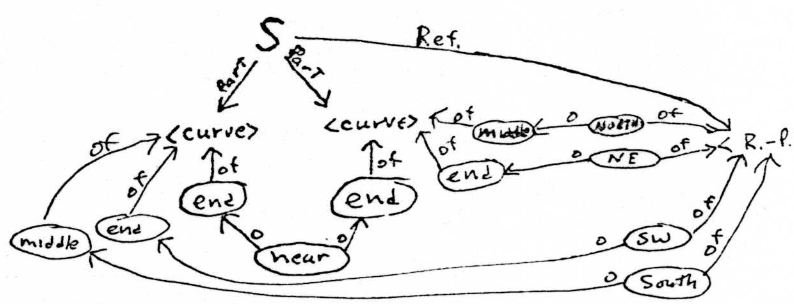


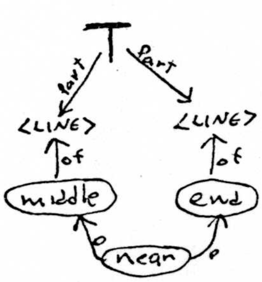


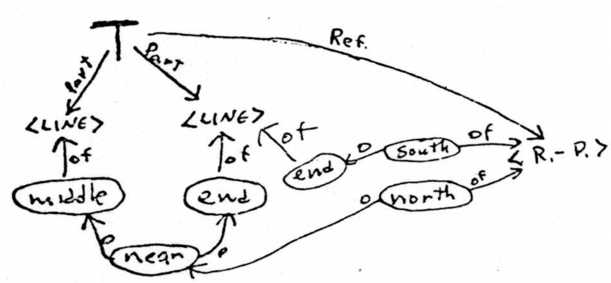






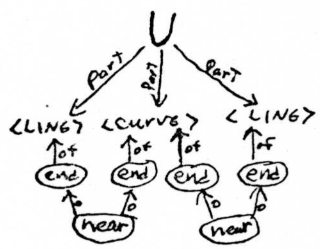


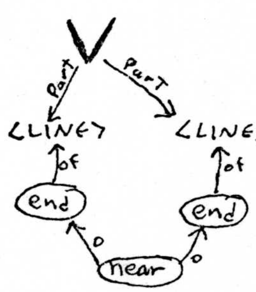


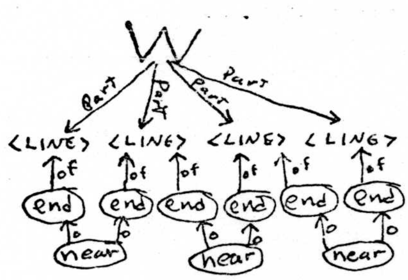


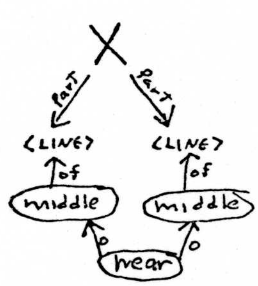
Orientation-free

OR

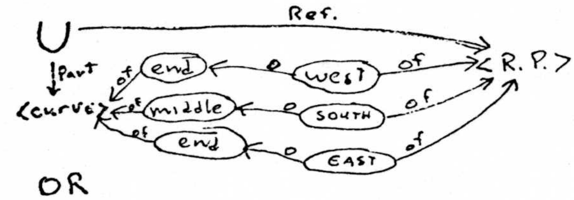




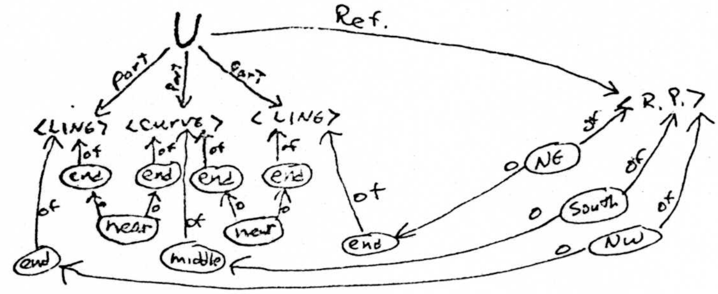


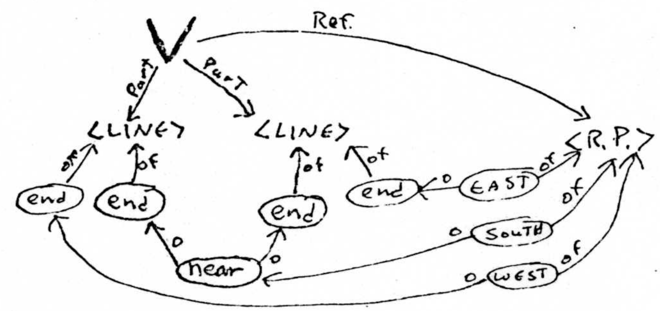


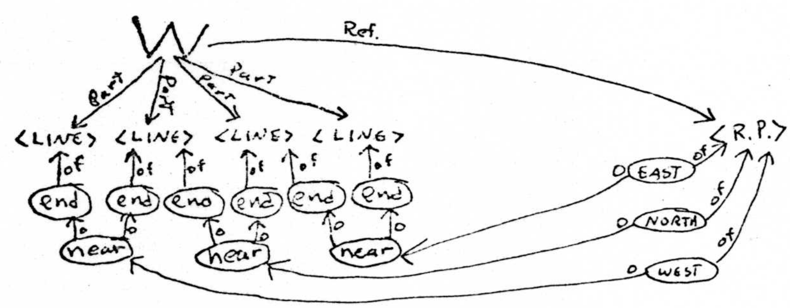
Orientation-sensitive

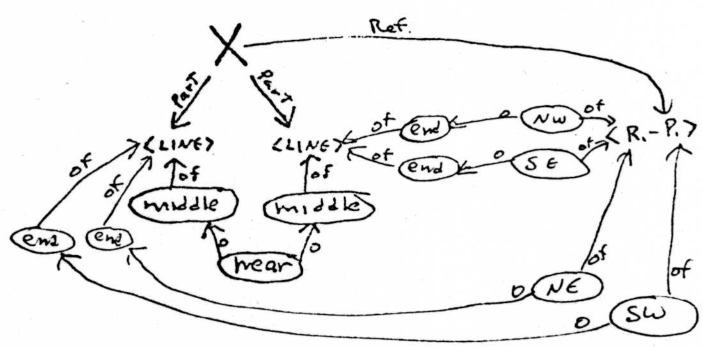


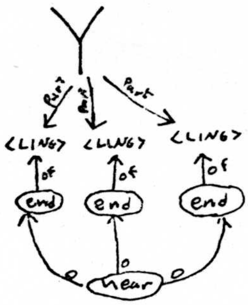
OR

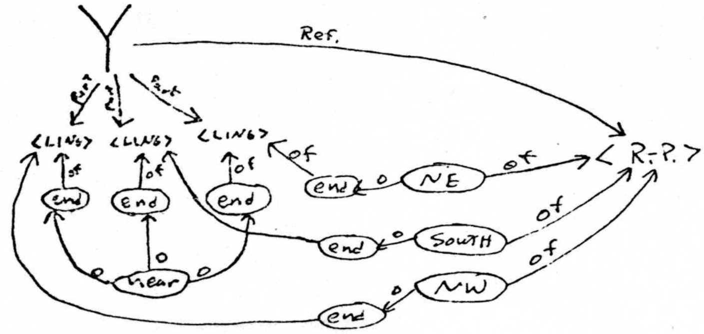






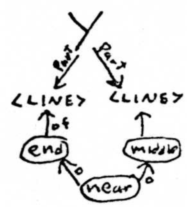


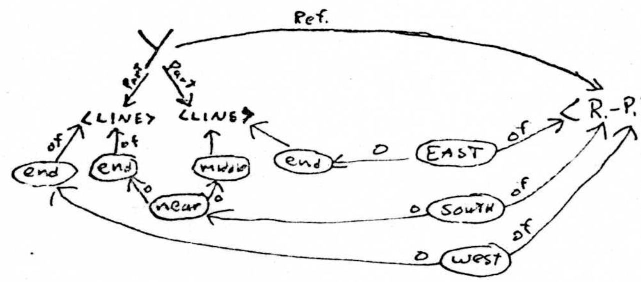


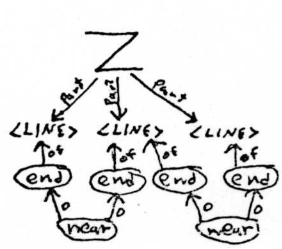


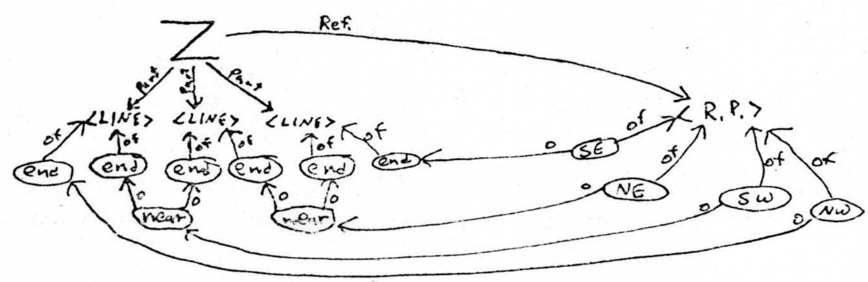
OR

OR









(Note: I don't claim psychological validity for these representations. The purpose of this appendix is only to demonstrate that the system I have described is, in fact, robust enough to represent all the letters. In fact, for some letters, several alternative representations will be presented.)