Spatial Configurations In The Model Invocation Process

Robert B. Fisher and Mark J. L. Orr

Department of Artificial Intelligence

University of Edinburgh

Abstract:

This paper presents a theory and some evaluations on how configuration evidence contributes towards model invocation. Invocation occurs when the plausibility of an image structure having a particular identity becomes sufficiently high. Plausibility is acquired from data properties that satisfy constraints and component relationships defined by the model. The data and model relationships define a large network within which plausibilities can be computed in parallel. This network can be conveniently mapped into registration with an image array, so that a processor can be designed that has all model base relationships defined permanently while still allowing dynamic network reconfiguration for each new image. Particular new results in this paper are:

(1) New data constraints are defined specifying evidence based on spatial configurations between features rather than feature property.

(2) Direct data evidence and subcomponent evidence are now integrated uniformly, under the assumption that both can be viewed as object features.

(3) The evaluation and network linkages for binary constraints are defined.

(4) The computation is defined for both orientation independent and viewer centered invocation.

Keywords: vision, model invocation, spatial configurations, network computations

1. Introduction

Interpreting image data as an instance of a named entity requires invoking the identity, whether to use as an index to other information or by itself as input into other visual processes. Invoking the model is a difficult problem because a competent visual system may know 10^4 - 10^5 nameable objects, yet all must be accessible quickly. Moreover, neither the object representations nor the model base will be complete, so the invocation process will have to invoke "close" models, such as when we see a familiar car with a new dent, or a new person.

Invocation is not a model matching process, rather, it is a suggestive process producing a small set of potential identities to explain the data. Hence, it may make errors, and need not depend on fully discriminating properties.

In [FIS85], a network formulation of the model invocation process was introduced. This process calculated the plausibility that a particular model was a potential explanation for a set of data, based on direct evidence from observed properties and indirect evidence from generic and component relationships. Invocation occurred when the plausibility of an image structure having a given identity reached a threshold level. The invoked identity is then usable for direct model matching, or for suggestive support in processing other visual data.

A computational account of this process has been developed ([FIS85])

specifying the invocation network structure, the functions calculated in the network and the mapping of the network to image data. One notable absence from the process was the representation and use of invocation cues available from the spatial configuration of features, as compared to feature types.

This paper extends the previous work to incorporate spatial configurations into the model invocation process. Starting with some intuitions on the role of configurations, it proposes a computational theory and presents some experimental results.

In the exposition below, surface and volumetric primitives will also be included as visual entities, because the invocation process may also be applied to image data already processed to the point of surfaces ([FIS85]) or volumes. Many of the examples will be taken from two dimensional configurations, for simplicity, but several examples at the end will demonstrate invocation of other structures. Alphabetic letters will be used as the primary source of spatial configurations. Some letters have similar configurations (e.g. H and I, d, b and p, 3 and S, L and J, etc.), and will therefore be a good test of our methods.

The work in this paper follows directly from the previous invocation research of Fisher ([FIS85]) summarized in section 3. Hinton ([HIN81a],[HIN81b]) proposed a network method integrating identity and orientation resolution - particularly for letter recognition - using tuned orientation units integrating at the component level. This paper takes a slightly different direction and suggests spatial relationships can be expressed implicitly between features (at less networking cost) and provide direct evidence for the configurations. The configurations invoked (for three dimensional objects) should be merely representative views (Minsky [MIN75]) and thus have a rough orientation implicit in them. This paper also explicitly

defines the types of linkages and functions in the network, based on properties of desired computations.

Feldman and Ballard ([FEL83]) proposed a discrete model of recognition based on spatially coincident properties. This was more properly a grouping of typed features (e.g. letters grouping to form words), but some configuration effects must be present because of the significance of letter order in word meaning. (Though it is hard to say whether the configuration is spatial or temporal.)

One key problem not fully resolved in any of these investigations is to what extent are the results of invocation suggestive, as compared to conclusive. The work here follows the view that it is suggestive (possibly strongly) and its results are then usable for model directed analysis.

The choice of object features and relationships strongly follow Marr ([MAR82]) and Sloman's POPEYE research group ([SLO78],[OWE80],[OWE82]). Marr and Nishihara ([MAR78]) proposed an axis oriented representation for solid objects with subcomponents linked by axis relationships. Owen ([OWE82]) used relative positions of side strokes about a main axis to describe configurations of strokes in hand printed letters. While many objects have axes, not all do and it is felt that integrating the evidence from local relations between features can make up for not having a global organizing axis. This approach can also be generalized for other than 3 dimensional elongated solids or letters. Owen ([OWE80]) argued that intermediate representations of letters (and words) should depend on two aspects: token identification and relative token placement. The relative placement aspect is the key to the configuration effect in invocation.

The indirect use of constraints on spatial relationships to recognize objects has been investigated several times. Barrow and Popplestone ([BAR71])

used the relations in arcs before graph matching, with weighting for the number of successful arc matches. Brooks ([BRO81]) implemented a more sophisticated version in ACRONYM, where an object had to meet the constraints to be considered for the identity. Barrow and Tenenbaum ([BAR76]), in MSYS, used spatial orderings to constrain region labelings. Adler ([ADL75]) and Hinton ([HIN76]) recognized 2D sketches of humans using parts configurations.

Here, the goal is not explicit representation or recognition, but instead to use the relationships as constraints contributing to model invocation.

Rock ([ROC74]) summarized various experiments on human perception of disoriented figures and the key results were:

(1) There is both an environmental and a retinal component of figure perception.

(2) Humans use environmental cues (e.g. gravity, scene verticals) to correct retinal orientations back to "vertical".

(3) The correction seems to dominate except when a strong axis is present, the figure is complex (e.g. a whole word) or the rotated version has a strong class resemblance to the unrelated form (e.g. cursive script).

(4) The key place descriptors seem to be roughly: "top", "bottom" and "side", but not "left" or "right".

Because of the evidence for the importance of environmental labeling, this factor is also included in the theory presented in chapter 4.

2. Intuitions and Motivations Behind Configurations In Invocation

The key motivations to include configurations into invocation come from:

(1) a group of features have a spatial distribution as well as feature types, and

(2) humans use cues from configurations.

This section will contain several examples motivated from human perception, but the goal is to illustrate what processes might be needed for a machine vision system, rather than understanding the human visual system. Configurations are a necessary aspect of invocation. It is clear from figure 1(a) that the line segments have no intrinsic relation to the triangle concept, so it must be their configuration (as in figure 1(b)) that invokes the triangle model.

(Figure 1)

The distinction between the use of feature types and feature locations in invocation is shown in figure 2. In (a), there are all the right components in the wrong places, and a face is invoked; this must be from type information. Cubist paintings (e.g. Picasso) exploit this process. In (b), the reverse occurs, with wrong feature types in typical places, and again the face model is invoked. Magritte's images often exploit this process.

(Figure 2)

The two processes complement each other. The first has already been incorporated into the invocation process ([FIS85]) and this paper develops the theory for the second.

If invocation evidence comes from token placement, what are suitable types of tokens? Figure 1 suggests line segments are one type, and figure 2 suggests point locations (e.g. the nose relative to the eyes) are another. Implicit segments through open space between points are also important, as seen in figure 3(a). Here, the dots suggest connecting straight line segments, again invoking the triangle. It may also be that the dots themselves map to the corners of the triangle and invoke through that path, but this

seems less likely in 3(b). We will assume that token points do no more than implicitly define line segments between them. Figure 4 shows this with two triangles invoked from segment endpoints. (Subjective contour phenomena probably also play a part here, but are ignored in this paper.)

(Figure 3)

(Figure 4)

Sequences of tokens also define line tokens, with the line passing through to the extrema, as seen in figure 3(b).

The axis of elongation 2D regions seem to be significant features, as figure 5 shows. Here, all three sketches are roughly equivalent, with the axes defining their character.

(Figure 5)

Arcs define tokens, as seen in figure 6. The arc implies two axes - one connecting the endpoints and the other giving the arc direction, the latter being necessary to distinguish a) from c). This may be a weaker constraint, as dyslexia appears to be fairly common. Correctly fixing the orientation could be done at higher levels.

(Figure 6)

Considering alphabetic letters as invokable configurations, the features defined so far are not sufficient to handle letters like O or Q. It may be the case that circular features define a point token at the center of the circle or, instead, the circle, being a nameable symbolic entity, should be used in the type based feature invocation process (e.g. figure 2(a)) rather than in the configuration process. This problem is not addressed further here, and the letters O and Q are omitted from the experiments discussed in section 5.

Configurations also occur in three dimensions. A tree can be treated as a blob at the end of a stick, a shoe as a collection of surfaces (figure 7). The two examples have the token types in a particular relationship to each other suggesting the desired object. The grouping of cylindrical solids to form a human body seems to be a distinctive configuration as well. The two key classes of tokens here seem to be surface patches and volumetric sticks, plates and blobs ([SHA80]).

(Figure 7)

Hence, the key token types include points, lines, arcs, surface patches and volumes. Since identities are not needed for the configuration based invocation process, this completes most of the primitive structural elements needed. What relationships between these features, then, define a configuration?

Previous research ([MAR82],[OWE80]) has suggested using a main axis with other components oriented about it. Marr's interest was in stable and canonical descriptions, so the axis provided a convenient organizational focus. The interest here is not in object representation, instead in using the spatial cues to suggest possible models. Further, not all figures have such an axis (e.g. the letter O and figures 1(b) and 7(b)). Symmetry also suggests an organizing axis, but this is not always the case, as in figure 8, where the symmetry axis seems minor compared to the elongation axis. Hence, this work will explore not using such an global organizer, but using only local feature pair relationships. This will define the global relations implicitly, much as a trinary predicate can be expressed as two binary predicates.

(Figure 8)

For sketch scenes, the key relationships between tokens are axis rela-

tionships, whether the axis is explicit in the token, or implicit between two point tokens. The three relationships used here are:

(a) relative axis orientation

(b) relative axis placement

(c) relative axis size

Marr, in defining model affixments ([MAR82]) links (a) and (b), but here they are separated because they provide different constraints on the token relationship. For the human stick figure, the relative orientation of the sticks is unimportant, compared to the attachment points and relative sizes, so each of these properties is expressed independently.

Parallelism is a possible feature, but it can be expressed by a relative orientation of zero between the axes. Feature containment is important for sketch scenes, where one feature is "inside" another, but this is ignored here.

For surfaces, relative surface orientation, size and placement are the key relationships, though adjacency might also be considered. As a surface token probably would not include a description of its extent, adjacency would not be inferrable from placement.

For solids, Shapiro et al ([SHA80]) give a good catalogue of relationships between stick like solids (end), plate-like solids (edge, interior) and blob-like solids (center):

> "The type of connection can be end-end, end-interior, end-center, end-edge, interior-center, or center-center ...".

Presumably edge-center was also intended.

The discussion so far has presumed an unoriented object, but this is often not the case, particularly with sketch entities such as letters. These are typically seen only in an "upright" position, so incorporation of viewer-

centered descriptors like "top" or "left", etc. would refine configuration descriptions, particularly if an object's identity was dependent on configuration, as in a square versus a diamond (figure 9). On the other hand, when most letters are upside down (e.g. an "R") it is still recognizably the same letter, so its orientation should not be an absolute requirement. Yet, an upside down "d" is usually seen as a "p", with the "d" never even considered. Rock ([ROC74]) suggested orientation is a strong factor in human perception.

(Figure 9)

Viewer centered relationships also occur with three dimensional objects - one seldom sees a human upside down, and a upside-down face is often not immediately identifiable.

Finally, there is the problem of which features should be grouped to form a configuration - what are the grouping principles. For sketches, two factors seem important - proximity and isolating structure. In figure 10, each individual set of features seems more tightly associated than features paired across groups, except when considering each group as a whole. In figure 11, the boundary seems to isolate the stars interior, even when then boundary is not convex.

(Figure 10)

(Figure 11)

Other factors include token type grouping (figure 12) and segmentation by recognition, so there are several isolating processes.

(Figure 12)

For surface and solid configurations, space occupancy constraints simplify the problem. Following [FIS85], the primitive connected surface groups

(isolated by connected obscuring and concave boundaries) create a context for inter-surface relations, and for volumes, the depth aggregated connected surface groups form complete solid objects. In both cases, being solids creates a natural context within which features lie.

Figure 13(a) shows a mug with a handle. The concave surface boundary segments the handle from the body and each of these forms a primitive connected surface group (figure 13(b)). Though the concave boundary is ambiguous regarding depth ordering, either alternative is equivalent and produces the whole cup for the depth aggregated connected surface group.

(Figure 13)

3. Review of the Invocation Formulation

Chapter 7 of [FIS85] presented a solution to the model invocation problem based on parallel computation of plausibility in a network. The key features of the formulation were:

- The nodes in the network represented every potential identity an image structure could acquire.
- Plausibility is an evaluation of how well the given model explains the corresponding image structure.
- The plausibility computation was based on functions implementing generic, component and direct evidence relationships.
- The arcs in the network link nodes related by the generic and component relationships within the appropriate image contexts.
- There is a natural mapping between the image contexts and the network, leading to a parallel computational structure.

These points are explained in greater detail below.

Each node in the network represents an identity that a corresponding im-

age structure might have. Suppose a simple domain contains five objects: a cylinder, a waste bucket and its three component surfaces - the inside cylindrical surface, the outside cylindrical surface and the bottom surface (the inside bottom has the same shape as the outside bottom). Suppose the scene is as in figure 14 below. Considering only surfaces and solid structures for the moment, this scene has four image structures of significance: the three surface regions (A,B and C) and the connected surface group consisting of surfaces A and B. The network for this scene has 11 nodes in it - 9 for the three surface identities each image surface could have, and 2 for each identity the connected surface group could have.

(Figure 14)

Each node in the network is associated with a plausibility value. This plausibility measure attempts to encode the potential of node's model as a correct explanation of its image structure. While plausibility is not a probabilistic measure, it does have the interpretation that higher plausibilities (for a given model) imply better agreement, and model invocation occurs for any node whose plausibility exceeds a given level.

Plausibilities are measured on a $[-1,1]$ scale, with negative values representing contradicting data and positive meaning supporting data. Values near zero are neutral, but invocation currently occurs for nodes with any positive plausibility.

A node acquires its plausibility from two types of evidence - direct and indirect. Direct evidence comes from properties of the data that meet constraints associated with the model identity, and the degree that these constraints are satisfied evaluates to a plausibility value. For example, the surface area of the waste bucket bottom is expected to be in the range from 160 to 240 cm^2, with a nominal value of 200. Then, for this property alone, a

data value of 220 cm^2 would contribute a plausibility value of 0.5, according to the function shown in figure 15. (All property evaluation functions have this shape.)

(Figure 15)

Each constrained property contributes a plausibility measure for the identity, and the overall direct evidence plausibility value is calculated from these. This calculation is a weighted average, where the weights scale the individual plausibilities according to the importance of the feature to the identity of the object, and the likelihood of obtaining reliable data. The averaging process treats negative plausibilities with twice the weighting of the positive plausibilities, because only a few properties may discriminate between the true and similar identities, and so any contradicting evidence is given greater significance. A typical portion of the network integrating the direct evidence evaluations for a node is shown in figure 16.

(Figure 16)

Constraint Evaluations

Indirect evidence comes from other nodes that have a given generic or component relationship with the current node. In our example, the waste bucket is treated as a specialization of the cylinder. A reasonable plausibility relationship between the two identities is that the plausibility of the specialization should be at most as large as that of the generalization. So, when computing the plausibility for an identity based on supertype evidence, one possible function would use the minimum of the plausibility values of its generalizations. In our example, only the connected surface groups have potential identities related by a generalization.

There are four indirect evidence relationships, mediated by different

computations. The four are: generalization, specialization, subcomponent and supercomponent. Each of these relationships has restrictions on the identities and the image structures (i.e. which nodes) which can contribute to the calculation of the final plausibility. Generic relationship plausibilities come from the same image structure, subcomponent plausibilities come from image structures contained within the current image structure and supercomponent plausibilities come from containing image structures. These restrictions define the links over which plausibility flows between nodes in the network.

Inhibitory inputs also add other links to the network. The main inhibition comes from other potential identities for the image structure. It is assumed that most structures are likely to have only one potential identity (disregarding generic relationships here), so an identity with a high plausibility should inhibit other identities for the same structure.

These different plausibility sources need to be integrated for each node. This also uses the positive minus twice negative averaging function.

(Figure 17)

Figure 17 shows a portion of the invocation network for computing the plausibility that the connected surface group is the waste bucket. The top left node represents the cylinder identity for the same image context, and (here) is a generalization of the waste bucket. The three small function units underneath it compute the generalization inputs for the integrated waste bucket plausibility, and (here) do not do much because only one generalization was defined. At the bottom of the diagram are six nodes for each of the three identities for the two surfaces contained in the waste bucket's image context. The square boxes above these compute the best plausibility for each of the different identities, which flow into the open circles, which compute plausibilities for three different groupings of the surfaces. Subcomponents are

seen in visibility groups depending on viewpoint, and each of the circles computes the plausibility for a given group. Above these, a square box selects the best grouping. In the test image, data surface A is an inner surface and B is an outer surface, so the rightmost subcomponent grouping should contribute the highest plausibility. At the left of the figure, inhibition inputs arrive from other competing identities, but for the example, no other identities are possible. Finally, at the center of the diagram, three units integrate all the different plausibility inputs to give a single value for the waste bucket, which may then be used as input into other computations.

This summarizes the processes in the network structure for computing plausibility. This formulation is ideal for parallel (and analog) execution, as each of the computations in the formulation can be executed in parallel, with values changing as new evidence is made available. What remains is to show how this structure can be mapped naturally onto the image data.

Figure 18 shows a series of planes spatially registered with the image. Assume:

(1) each of the upper planes represents a given model identity,

(2) each plane is a two dimensional set of processors linked with their adjacent neighbors, such that all such linked processors compute the same value (i.e. plausibility), and

(3) the surface and connected surface group boundaries from the two bottom planes (i.e. the image) define a set of vertical cutting surfaces that sever interprocessor connections.

Then, the processors isolated within each section of each plane correspond to the nodes defined previously, and internode connections go directly vertically within the enclosing contexts. Hence, the invocation network can be statically defined for all known objects, but also dynamically reconfigured for each new image. (This design is not intended to imply that each processor neccs-

sarily represents a literal physical component, but only that each is a unit of computation.)

(Figure 18)

We have now introduced most of the structure of the current invocation model, except that there are also nodes for line-type identities, which are needed for interpreting line oriented scenes, such as drawings or written text. Our formulation integrates both direct evidence and associations, is capable of fast operation through parallel implementation, is incremental, is tolerant to data errors, and selects models based on similarity, so can handle previously unknown, but similar data.

4. Theory of Configurations in Invocation

Following the intuitions given in section 2 and the review of previous results in section 3, some constraints can be given to guide the theory proposed in this section. They are:

On evidence:

- Evidence is incremental (i.e. more evidence should increase certainity of identity).
- Evidence is not perfect.
- Each piece of evidence should contribute to the final plausibility.
- The contribution of a piece of evidence should be a function of the degree to which it meets its constraints.
- Negative evidence should have a greater effect than positive evidence.
- Evidence comes from both direct data and indirect (i.e generic or component) relationships.
- Evidence comes from token spatial relationships as well as types.

On model constraints:

- All modeled constraints require evidence (occlusion is not considered

here).

- All constraints for all generalizations of the identity also apply.
- The better a data constraint is satisfied, the higher its contribution to an object's plausibility.

On data descriptions:

- Every description must meet a constraint, if any of the appropriate type exist.
- Not all description types are constrained (i.e. some properties are irrelevant).

Several other constraints on the computation have been used previously, but here, while we are concentrating on configuration evidence, are omitted for simplicity:

- Constraints have relative importance.
- The contribution of a piece of evidence should be a function of its importance in uniquely determining the object.
- Each piece of evidence should be considered only for the best fitting constraint.
- Not all constraints need evidence (e.g. because of occlusion, alternate viewpoints).

Having presented our guiding intuitions, the following configuration invocation theory is proposed.

Configuration Features

As discussed in section 2, several visual features create tokens for the configuration process. They are defined:

point - a distinguished spatial location, such as: the end of a line, the center of a circle, the center of mass of a surface or a volume, or

the nominal location of a previously recognized object. Its property is location.

segment - connects the endpoints of a sequence of points through the sequence (type 1), connects two points through open space (type 2) or is the elongation axis of a narrow region (type 3). A segment may be curved or straight; curves cannot be type 2. A connected curve should be segmented at significant orientation or curvature discontinuities or when curvature reverses direction. The properties of the segments are:

(1) direction

(2) size

(3) location

If curved, then:

(4) direction of curvature.

The details of how these are represented is ignored for the moment. The magnitude of the curve does not seem to be important.

surface - a segmented portion of the complete object surface. Segmentation criteria might be based on depth, surface orientation and surface curvature discontinuities ([FIS85]). Its properties are:

(1) nominal surface normal

(2) size

(3) location

(4) curvature directions

volume - a segmented portion of a whole object. Segmentation criteria might be at connected concave boundaries ([FIS85]), radical changes in the orientation of elongation axes or volume joins ([NEV77]). Its properties are:

(1) orientation (if essentially 1D or 2D)

(2) size

(3) location

Configuration Relationships

The essence of configurations lies not in the individual tokens, but rather in their interrelationships. These are the properties evaluated in the configuration invocation process. The properties are defined similarly for the model base and the data, allowing direct comparison. For 2D sketches, segments are the key tokens, and their properties are:

(1) LINE

or

CURVE

which describe the character of the segment,

(2) RELSIZE <segment_1> <segment_2> <size_ratio>

giving the ratio of segment sizes of segment_2 as a proportion of segment_1, and

(3) AXORT <segment_1> <segment_2> <orientation>

giving the interior angle between the two segments. Figure 19 illustrates these cases. The point of using the interior angle is to distinguish between the relationships in figure 19(a) and 19(b), which have the same angles in the intersection of their extensions. Crossing or touching segments have both angles represented (figure 19(c)). If a segment is curved, the axis lies across the curve instead of through the endpoints (figure 19(d)).

(Figure 19)

(4) PLACE <segment_1> <segment_2> <distance_1>

defines the placement relationship between the two segments. The distance is normalized by the lengths of the segments and represents the length along the first axis where the extension of the second crosses. Figure 20 illustrates these constraints. (Values v and 1 - v are equivalent.)

(Figure 20)

(5) VIEWORIENT <orient>

defines any viewer-centered orientation requirements, with angles measured from the vertical (only in the range $[-\pi/2, \pi/2]$). Other viewer centered configuration descriptions and their meanings are:

(6) TOP <feature> feature appears in top half of configuration

or

SIDE <feature> feature appears on either side of configuration

or

MIDDLE <feature> feature appears in middle of configuration

or

BOTTOM <feature> feature appears in bottom half of configuration

which describe where in the oriented configuration a segment occurs. The directions are defined with respect to standard vertical appearance. These constraints require the corresponding data to have the same location constraints as the model. For example, given "TOP f2", to evaluate a relation

involving f2 (e.g. R(f2,f1)) with some data x paired with f2 requires also having "TOP x".

To specify what features belong together in a configuration, the following declaration is needed:

(7) CONFIG <config_name> <component_1> ... <component_n> ENDCON

(Figure 21)

Putting these together, a description of the configuration for the letter "R" (figure 21) is:

```
COMMENT R
CONFIG R r1 r2 r3 ENDCON

COMMENT r1
LINE r1
RELSIZE  r1  r2  2.0
RELSIZE  r1  r3  1.40
AXORT  r1  r2  1.57
AXORT  r1  r3  0.79
AXORT  r1  r3  2.35
PLACE  r1  r2  0.25
PLACE  r1  r3  0.5
VIEWORIENT  r1  0.0
SIDE r1

COMMENT r2
CURVE r2
RELSIZE  r2  r1  0.5
```

```
RELSIZE  r2  r3  0.7
AXORT    r2  r1  1.57
AXORT    r2  r3  0.79
PLACE    r2  r1  0.0
PLACE    r2  r3  -0.5
VIEWORIENT  r2  1.57
TOP      r2

COMMENT r3
LINE r3
RELSIZE r3 r1  0.7
RELSIZE r3 r2  1.4
AXORT    r3  r1  0.79
AXORT    r3  r1  2.35
AXORT    r3  r2  0.79
PLACE    r3  r1  0.0
PLACE    r3  r2  -0.5
BOTTOM   r3
VIEWORIENT  r3  -0.79
```

For surfaces and volumes, the axis orientation and relative size relationships are analogous, so the same predicates will be used. Though 3D placement relationships are more complicated, physical constraints will allow simpler descriptions. In part, the constraints are that objects are solid and must be fully connected. The new descriptor for surfaces is:

(8) SOLIDANGLE <surface_1> <surface_2> <angle>

if the two surfaces are adjacent. A pair of non-adjacent surfaces so described are not fully specified by this, but by object solidity, they have

connecting intermediate surfaces that constrain them. Surface normals relate to axes and surface areas give relative sizes, so previous predicates are still usable.

For solids, the new descriptors are:

(9) TYPE <solid> <type>

where <type> is STICK, PLATE or BLOB, and

(10) CONNECT <solid_1> <solid_2> <connect>

where <connect> is: END_END, END_INTERIOR, END_CENTER, END_EDGE, EDGE_CENTER, INTERIOR_CENTER, or CENTER_CENTER. Most previous descriptors are still usable here, too.

These constraints are designed to express significant relations in configurations, not all relations, so there are some relations that will not be constrained explicitly, or perhaps even implicitly. The point is not to provide discriminative power, but suggestive invocation.

Evaluating Individual Constraints

Given the data values and constraints, the problem is to calculate an initial plausibility. For unary constraints, like "VIEWORIENT r1 0.0", the data evaluation function of figure 15 is used (following [FIS85]), where:

For property i evaluated with data from feature j, let:

$$d_{ij} = |data_j - mean_i| = |(data_j - mean_i) / mean_i|$$

Then:

$$unary\_plaus_{ij} = \text{if } (d_i > \tau_i) \text{ then } -1$$

else $1 - 2*d_{ij}/\tau_i$

The tolerances τ_i allow for data errors and variation within a class. As implemented, the tolerances were:

i	τ_i
RELSIZE	0.2
AXORT	0.2
PLACE	0.2
VIEWORIENT	0.2
SOLIDANGLE	0.2

Thus we can evaluate the plausibility of a single data feature satisfying a single model constraint. Our interest, however, is in how well the model constraint can be satisfied, i.e. what is the best satisfaction of the constraint. This means all possibilities should be considered, and the best evaluation chosen. For each model constraint, all data for that constraint type is considered. This seems like weak discrimination, but recall that at this point no model has been selected, and so model-data correspondences that would expose contradictions have not yet been made.

The computation computing the plausibility for $constraint_i$ over all $data_j$ measurements is:

$$constr\_plaus_i = max_j \ (\ unary\_plaus_{ij} \ )$$

For binary constraints, like "RELSIZE r1 r2 1.0", the plausibility of the second feature must be considered. Here, we want the plausibility to be a function of both how well the property is satisfied and the likelihood that the second feature is the desired item. The intuitive guidelines for this

computation are somewhat uncertain, but obviously: (1) if either the value constraint or the identity is weak, then the final result should weak too, and (2) if both are strong, then the result should be strong. The heuristic chosen for this process is the function comb given below. The computation for the binary $constraint_i(f_i,f_k)$ is then:

$$constr\_plaus_{ik} = max_j \; (\; comb \; (\; unary\_plaus_{ij} \; , \; id\_plaus(d_j = f_k) \;) \;)$$

where:

d_j are the potential second data features,

id_plaus(d = f) is the plausibility that data feature d has the model feature identity f, and

$$comb(x,y) = 1 - (1 - x) * (1 - y)$$

The non-numerical constraints (e.g. CONNECT) contribute 1.0 if there is any data satisfying the constraint, and -1.0 if not.

This version of data constraint evaluation does not take viewer orientation into account, hence a data "R" at any orientation should achieve the same evaluation. Viewer centered computations were also implemented, and how this modifies the above formulation is now discussed.

In evaluating each of the above constraints, any data feature could currently be paired with a model feature. In the viewer centered implementation, only data features having the same configuration location property (i.e. TOP, BOTTOM, SIDE, MIDDLE) are allowed to be paired. Further, the VIEWORIENT constraint is evaluated only in the viewer centered formulation. The use of viewer centered data has the effect of sharpening the distinctions between configurations (such as "I" and "H").

Low Level Configuration Network Formation

The constraint evaluation described above provides the first layer in the invocation network, with each model constraint selecting its maximum evaluation for all data within its context. In network terms, this creates a linkage from all evaluations of the constraint on the data to a function picking the maximum. Figure 22 shows a typical fragment from this portion of the network.

(Figure 22)

Modeled relationships also contribute to plausibility. The above constraints express only local pairing relationships between features. For the configuration as a whole, its plausibility also depends on the the plausibility of its subcomponents at achieving their identities, as well as any other constraints that might be specified (e.g from participation in larger configurations). Hence, subcomponent existence is treated as a new property, and the plausibility contribution for each subcomponent is the maximum of the plausibilities of each data feature being that subcomponent. (An extra constraint that could have been used was that a data feature could contribute plausibility to only one model feature, but this was felt to be unnecessary.) The subcomponent relationship provides a network linkage from individual features to the configuration.

Individual features also acquire plausibility by participation in configurations, from the plausibility of the supercomponent. This becomes a new constraint, whose evaluation is the plausibility of the data configuration having the desired model configuration type. This also defines a new network linkage.

There is also an inhibition constraint. The rationale for this is that

an object should not receive high plausibilities when there are several competing identities. The computation for this picks the highest plausibility among all specified competitors for the given type. If this is positive, then its negative is used as a negative constraint (weighted with double strength). If is negative, then it is ignored, as negative evidence for one type is no support for another type. The effect of this is to sharpen plausibility differences and force weak plausibilities into implausibility in the face of stronger identities. If no direct evidence is obtained, no inhibition is applied.

To get the final plausibility value for the potential of the identity to explain the data, the individual constraint plausibilities are integrated by the following procedure:

```
weight = 0, sum = 0

FOR (<each constraint plausibility p>) {
    IF p >= 0
        ADD p TO sum
        ADD 1 TO weight
    ELSE
        ADD 2*p TO sum
        ADD 2   TO weight
}

final_plausibility = sum / weight
```

The doubled negative weights place greater emphasis on contradicting evidence.

The network fragment that summarizes these points for the invocation of the "R" configuration defined above is shown in figure 23. At the top is a rectangular node recording the plausibility of the data configuration D

achieving the model identity R. It receives this plausibility from the circle/plus integration unit directly below it. This unit integrates the subcomponent evidence with the inhibition inputs. The inhibition comes from all other potential identities for the configuration D. The squares below compute the evidence for the defined subcomponents by picking the maximum plausibility of each of the three data elements having the desired identity.

Node $d_2 = r_2$ is partially expanded below. Again there is the circle/plus integration unit with the inhibition unit to its left. Here one inhibition input is shown explicitly, with the other inputs coming from the other competing identities for data d_2. The integration is shown for three properties among those defined above. At the right is a maximum unit picking the best supercomponent evidence, here from only the D = R node. (A feature may be used in several supercomponents.) Next to the left is an evaluation of the unary "VIEWORIENT" constraint with inputs from the data for feature d_2. At the bottom left is a unit computing the binary plausibility for the "RELSIZE" constraint. It takes the maximum of a function of the RELSIZE evaluation for all features having this relation to feature d_2. The other input to the function is the plausibility that the second feature has the correct identity (here r_2), which comes through the links from the nodes defined near the middle of the network.

(Figure 23)

Network Computation

The above sections defined the network structure relating plausibility values to model definitions, evidence relationships and property evaluations. To compute the plausibility of any node, the network is evaluated iteratively until steady state, where all values are consistent, as defined by the data, network structure and the computations proposed for each functional unit.

In practice, networks were occasionally found having one or two isolated nodes which oscillated in plausibility between successive iterations. An updating function incorporating averaging removed most occurrences of this, at the expense of slower convergence.

Invocation occurs for data/model pair nodes acquiring a positive plausibility evaluation.

Evidence Contexts

As discussed in sections 2 and 4, evidence comes from within contexts:

- 2D image boundaries: the clustering and isolation boundary processes determine which data associate. Relations only hold between nearby features. Implicit segments crossing explicit segments are ignored.
- 3D boundaries: the segmentation boundaries isolating a surface are the data. Relations only hold between adjacent features.
- 3D surfaces: the connected surface groups isolate these. Relations only hold between adjacent features.
- 3D volumes: the depth aggregated connected surface groups isolate these. Relations only hold between adjacent features.

This plausibility evaluation process can again be implemented in a parallel network similar to figure 18. The surface and volume contexts above carry over directly under the original theory. For the 2D boundary configurations, isolating boundaries are postulated and arise from either literal boundaries or decay effects over open space. Both processes effectively sever horizontal interprocessor connections. For 3D boundary connections, the information flows only through processors lying on the boundaries.

Constraint evaluation units tuned for typical values make up the lower, identity independent levels of the network - thus allowing for low-level sym-

bolic vocabulary sharing among more complex structures.

5. Experiments with Configurations

Several experiments were done to evaluate the theories proposed in the previous section. The simplest is more of a demonstration and calculates the plausibilities for a face model using a good and bad face configuration, as suggested by figure 1. Appendix A shows the models and the faces are in figure 24. The features defined were the line defined by the eye tokens, the nose and the mouth. The plausibilities for the good and bad data as instances of the good model when no viewer centered frame is required are:

good head data	bad head data
0.51	-0.63

The results when the viewer frame is required are stronger:

good head data	bad head data
0.99	-0.86

This shows the configuration evidence is significant, the network discriminates and that the viewer reference frame enables stronger discrimination. One criticism of the test is the theory allows other implicit lines between feature points (e.g. end of mouth and eye), which were not incorporated in either data set. The net effect of this would probably not be significant - though the presence of coincidental property matches is more likely, there are also more that require satisfaction.

(Figure 24)

A larger example demonstrates the process more carefully. Configurations were defined using alphabetic characters (except G, O and Q). A network evaluation using a subset of these was performed, with the summary of results

shown in appendix B. The subset was selected because of similarities between characters, and discriminations involving the other characters not in this set were always perfect. The network formed using these 11 configurations against themselves contained 1600 nodes.

In the non-viewer centered case, the correct model was always invoked when applied to the correct data. Most of the wrong pairings had negative plausibility and so were not invoked. Those wrong pairings which were invoked were actually quite reasonable, e.g. the F model on the E data. Some points to notice about the results are:

(1) The E data applied to the F model invokes, as is expected, but the E model has higher plausibility, because it has more satisfied properties than the F model. Inhibition of evidence emphasizes the differences.

(2) The H and I data on the opposite models have neutral plausibilities, as their shape is similar, ignoring orientation. The same is true for L and T.

(3) The same holds for configurations with curves: B, P and R against each other also acquire neutral plausibilities (but still negative).

The same experiment was run requiring viewer centered invocation. Then, stronger invocations were achieved (more positive main diagnal of the results matrix) and larger suppression of the other pairings. Some points to notice about the results are:

(1) H and I are now strongly distinguished.

(2) The E model on the F data invocation is now suppressed (because the lowest horizontal line on the E has to be "BOTTOM" not "TOP").

(3) A side effect here is that the additional satisfied viewer centered properties now give small positive plausibilities to formerly negative plausibilities near zero. This means an L is now invoked with E data and a P is invoked with B data, but both of these cases seem

reasonable since the features overlap.

A final simple example shows good performance using three dimensional configurations. Here, a robot is invoked against a human model. Figure 25 shows these objects. No occlusion is assumed here, but aspects of this problem were considered in [FIS85].

The result for the non-viewer centered frame was:

		data	
		robot	person
model	robot	0.70	-0.43
	person	-0.37	0.39

For the viewer centered case, the result is:

		data	
		robot	person
model	robot	0.99	-0.48
	person	-0.57	0.68

These results seem to support the conclusions drawn from the alphabet experiment, namely that configurations are a useful source of evidence, and that viewer centered information adds to the discriminatory power

(Figure 25)

6. Discussion

The theory presented above is not purely based on configurations, in that subcomponent evidence is used to invoke the configuration itself. The spatial constraints constrained the possible interpretations of features locally, and

the object context provided context support and integrated the subcomponent evidence. The specific properties were not important, but adding more of various types (1) gives stronger general discriminatory power and (2) introduces new competences, such as distinguishing between figures 2(a) and 2(b).

One minor flaw in the implementation is the inability to distinguish curve orientation, as between a "3" and an "S". The solution is to add a new configuration property, though another possibility is that the distinction will be made during model directed inspection rather than in invocation. Dyslexia suggests that some orientation factors are weaker.

The letter recognition example relates to word recognition, but then other factors need be considered. Here, the primitive types contributing to a configuration were only weakly typed (only CURVE or LINE, instead of "F" or "A"). For words, the features are letters, which are strongly typed, so a theory based more on type and component relationships ([FIS85]) should be used. Further, the ordered scanning of the word during reading creates a temporal token configuration for which no theory is proposed here. (Feldman and Ballard discuss this in [FEL84].) There is likely to be substantial topdown support for letter identity during word recognition, given the distinctness of most words. Finally, we suggest that word recognition might be achievable through only invocation and not also model directed inspection, hence permitting "gestalt" like comprehension and fast reading competence.

This research leaves many open questions. Among these are the questions of how scale affects the grouping that makes a configuration a significant identifiable entity, instead of merely a feature in a larger configuration. When the possibility of different spatial frequency channels is allowed, feature blurring may reduce an identifiable feature to a point definition in a larger configuration.

Marr ([MAR82]) suggested numerical quantities should be scale quantified and symbolic rather than analog. This allows symbolic comparison of property values and we feel that this is probably a better direction to follow than the numerical evaluation used here. The question is how to quantify these and evaluate matches. For relative size, it may be something like: "much larger", "larger", "similar", "smaller" and "much smaller", with evaluation depending on the category closeness.

Another question is on the number of features in a configuration before binary relationship combinatorics cause all configurations to appear similar (we anticipate about 7, for obvious reasons). Finally, it seems likely that there is a large low level vocabulary of features shared among descriptions. The richer the vocabulary, the fewer relationships needing expression to formulate a new configuration, hence the more complex configurations expressible. The question is over what are the general types of primitives and how are they represented. (This seems like a general cognitive problem, as does invocation.)

This paper presented a theory and some evaluations on how configuration evidence contributes towards model invocation. Invocation occurs when the plausibility of an image structure having a particular identity becomes sufficiently high. Plausibility is acquired from data properties that satisfy constraints and component relationships defined by the model. Here, only configuration and component constraints were considered, but in a complete invocation process, other direct and indirect evidence constraints would also be integrated. The data and model relationships define a large network within which plausibilities can be computed in parallel. It was shown that this network can be conveniently mapped into registration with an image array, so that a processor can be designed that has all model base relationships defined permanently while still allowing dynamic network reconfiguration for each new im-

age. (Though many other implementations are also possible).

Some of these results have been presented elsewhere. New results this paper presents are:

(1) New data constraints were defined specifying evidence based on spatial configurations between features rather than feature property. These properties were mainly binary and local, and so defined more complicated configurations implicitly.

(2) Direct data evidence and subcomponent evidence were integrated uniformly, under the assumption that both can be viewed as object features. This simplified the network formulation.

(3) The evaluation and network linkages for binary constraints were defined. Previously, only unary constraints were used.

(4) The computation was defined for both orientation independent and viewer centered invocation, and was found to have stronger invocation behaviour in the viewer centered case.

7. References

[ADL75] Adler, M., "Understanding Peanuts Cartoons", Edinburgh DAI Res. Rpt. #13, 1975.

[BAR71] Barrow, H. G., Popplestone, R. J., "Relational Descriptions in Picture Processing", Meltzer & Michie, "Machine Intelligence 6", pp377-396, 1971.

[BAR76] Barrow, H. G., Tenenbaum, J. M., "MSYS: A System For Reasoning About Scenes", SRI Technical Note 121, 1976.

[BRO81a] Brooks, R.A., "Symbolic Reasoning Among 3D Models And 2D Images", Stanford AIM-343, STAN-CS-81-861, 1981.

[FEL83] Feldman, J. A., Ballard, D. H., "Computing With Connections", in "Human and Machine Vision", Beck, Hope and Rosenfeld (eds), Academic Press, pp 107-155, 1983.

[FIS85] Fisher, R. B., "Recognizing Objects Using Surface Information And Object Models", University of Edinburgh, PhD thesis, forthcoming.

[HIN76] HintonG, "Using Relaxation To Find A Puppet", AISB-76, 1976.

[HIN81a] Hinton, G. P., "A Parallel Computation That Assigns Canonical Object-Based Frames of Reference", Proc. IJCAI 7, pp683-685, 1981.

[HIN81b] Hinton, G. P., "Shape Representation In Parallel Systems", Proc. IJCAI 7, pp 1088-1096, 1981.

[MAR82] Marr, D., "Vision", pubs: W.H. Freeman and Co., 1982.

[MAR78b] Marr, D., Nishihara, H. K., "Representation and Recognition of the Spatial Organization of Three Dimensional Shapes", Proc. Royal Soc., Vol. 200, 1978.

[MIN75] Minsky, M., "A Framework For Representing Knowledge", in Winston (ed) "The Psychology of Computer Vision", McGraw-Hill, 1975.

[NEV77] Nevatia, R., Binford, T. O., "Description and Recognition Of Curved Objects", AI Vol. 8, pp77-98, 1977.

[OWE80] Owen, D., "Intermediate Descriptions In "POPEYE"", AISB-80, July 1980.

[OWE82] Owen, D., "Relating Object Models To Segmentation Strategies", Proc. ECAI 82, 1982.

[ROC74] Rock, I., "The Perception of Disoriented Figures", in "Image, Object and Illusion", readings from Scientific American, Held (ed), pp71-78, 1974.

[SHA80] Shapiro, L., Moriarty, J., Mulgaonkar, P., Haralick, R., "Sticks, Plates, And Blobs: A Three-Dimensional Object Representation For Scene Analysis", AAAI-80, Aug 1980.

[SLO78] Sloman, A., Owen, D., Hinton, G., Birch, F., O'Gorman, F., "Representation And Control In Vision", Proc. AISB 78, pp309-314, 1978.

Appendix A - Face models

GOOD FACE MODEL

```
COMMENT face
CONFIG face eyeeye nose mouth ENDCON

COMMENT eyeeye - line between eyes
LINE eyeeye
RELSIZE eyeeye nose 1.5
RELSIZE eyeeye mouth 1.0
AXORT eyeeye nose 1.57
AXORT eyeeye mouth 1.57
PLACE eyeeye nose 0.5
PLACE eyeeye mouth 0.5
VIEWORIENT eyeeye 1.57
TOP eyeeye

COMMENT nose
LINE nose
RELSIZE nose eyeeye 0.66
RELSIZE nose mouth 0.66
AXORT nose eyeeye 1.57
AXORT nose mouth 0.0
PLACE nose eyeeye 0.1
VIEWORIENT nose 0.0
MIDDLE nose

COMMENT mouth
CURVE mouth
RELSIZE mouth eyeeye 1.0
RELSIZE mouth nose 1.5
AXORT mouth eyeeye 1.57
AXORT mouth nose 0.0
```

```
PLACE mouth eyeeye 4.0
VIEWORIENT mouth 0.0
BOTTOM mouth
```

BAD FACE MODEL

```
COMMENT face
CONFIG face eyeeye nose mouth ENDCON

COMMENT eyeeye - line between eyes
LINE eyeeye
RELSIZE eyeeye nose 2.0
RELSIZE eyeeye mouth 1.33
AXORT eyeeye nose 0.70
AXORT eyeeye mouth 0.65
PLACE eyeeye nose 0.1
PLACE eyeeye mouth 1.7
VIEWORIENT eyeeye 0.4

COMMENT nose
LINE nose
RELSIZE nose eyeeye 0.5
RELSIZE nose mouth 0.66
AXORT nose eyeeye 0.70
AXORT nose mouth 0.65
PLACE nose eyeeye 5.0
VIEWORIENT nose -0.3
SIDE nose

COMMENT mouth
CURVE mouth
```

RELSIZE mouth eyeeye 0.75

RELSIZE mouth nose 1.5

AXORT mouth eyeeye 0.65

AXORT mouth nose 0.65

PLACE mouth eyeeye 7.0

VIEWORIENT mouth -0.1

TOP mouth

Appendix B - Alphabet Configuration Evaluations

NOT VIEWER CENTERED

DATA

		A	B	E	F	H	I	L	P	R	S	L
	A:	0.49	-0.62	-0.44	-0.41	-0.55	-0.55	-0.53	-0.61	-0.63	-0.65	-0.55
	B:	-0.60	0.37	-0.15	-0.25	-0.38	-0.31	-0.36	-0.13	-0.23	-0.45	-0.45
	E:	-0.56	-0.22	0.27	0.18	-0.20	-0.12	-0.28	-0.50	-0.42	-0.55	-0.34
M	F:	-0.53	-0.32	0.08	0.12	-0.26	-0.26	-0.19	-0.41	-0.35	-0.62	-0.27
O	H:	-0.51	-0.33	-0.05	-0.14	0.37	-0.05	-0.18	-0.29	-0.30	-0.54	-0.30
D	I:	-0.51	-0.34	-0.01	-0.09	-0.10	0.38	-0.21	-0.41	-0.40	-0.54	-0.17
E	L:	-0.49	-0.31	-0.09	-0.03	-0.19	-0.20	0.16	-0.25	-0.35	-0.59	-0.13
L	P:	-0.59	-0.13	-0.29	-0.24	-0.31	-0.40	-0.23	0.24	-0.20	-0.52	-0.34
	R:	-0.64	-0.48	-0.41	-0.38	-0.47	-0.53	-0.45	-0.44	0.49	-0.67	-0.52
	S:	-0.59	-0.18	-0.32	-0.46	-0.34	-0.35	-0.60	-0.47	-0.59	0.11	-0.63
	T:	-0.51	-0.40	-0.17	-0.12	-0.25	-0.15	-0.10	-0.34	-0.35	-0.59	0.24

VIEWER CENTERED

DATA

MODEL		A	B	E	F	H	I	L	P	R	S	L
	A:	0.83	-0.72	-0.50	-0.65	-0.64	-0.85	-0.91	-0.91	-0.85	-1.00	-0.93
	B:	-0.89	0.38	-0.07	-0.68	-0.84	-0.67	-0.75	-0.57	-0.49	-0.73	-0.94
	E:	-0.77	-0.38	0.40	-0.42	-0.71	-0.68	-0.73	-0.81	-0.71	-0.89	-0.90
	F:	-0.67	-0.50	0.19	0.85	-0.58	-0.79	-0.87	-0.71	-0.63	-0.97	-0.87
	H:	-0.56	-0.63	-0.28	-0.39	0.76	-0.82	-0.83	-0.83	-0.76	-1.00	-0.91
	I:	-0.84	-0.54	-0.41	-0.81	-0.81	0.81	-0.87	-0.91	-0.84	-0.82	-0.58
	L:	-0.76	-0.16	0.02	-0.75	-0.68	-0.72	1.00	-0.82	-0.56	-1.00	-0.89
	P:	-0.82	0.04	-0.17	-0.33	-0.74	-0.78	-0.88	1.00	-0.24	-0.88	-0.89
	R:	-0.87	-0.42	-0.49	-0.70	-0.83	-0.84	-0.86	-0.63	0.83	-0.94	-0.92
	S:	-0.90	0.00	-0.19	-0.84	-0.86	-0.37	-0.93	-0.78	-0.67	1.00	-0.93
	T:	-0.79	-0.62	-0.56	-0.73	-0.75	-0.23	-0.89	-0.89	-0.79	-0.94	1.00