Communicated by Geoffrey Hinton

Optimization in Model Matching and Perceptual Organization

Eric Mjolsness
Department of Computer Science,
Yale University, New Haven, CT 06520, USA

Gene Gindi
Department of Electrical Engineering,
Yale University, New Haven, CT 06520, USA

P. Anandan
Department of Computer Science,
Yale University, New Haven, CT 06520, USA

We introduce an optimization approach for solving problems in computer vision that involve multiple levels of abstraction. Our objective functions include compositional and specialization hierarchies. We cast vision problems as inexact graph matching problems, formulate graph matching in terms of constrained optimization, and use analog neural networks to perform the optimization. The method is applicable to perceptual grouping and model matching. Preliminary experimental results are shown.

1 Introduction

The minimization of objective functions is an attractive way to formulate and solve visual recognition problems. Such formulations are parsimonious, being expressible in several lines of algebra, and may be converted into artificial neural networks which perform the optimization. Advantages of such networks including speed, parallelism, cheap analog computing, and biological plausibility have been noted (Hopfield and Tank 1985).

According to a common view of computational vision, recognition involves the construction of abstract descriptions of data governed by a database of *models*. Abstractions serve as reduced descriptions of complex data useful for reasoning about the objects and events in the scene. The models indicate what objects and properties may be expected in the scene. The complexity of visual recognition demands that the models be organized into compositional hierarchies which express object-part

relationships and specialization hierarchies which express object-class relationships.

In this paper, we describe a methodology for expressing model-based visual recognition as the constrained minimization of an objective function. Model-specific objective functions are used to govern the dynamic grouping of image elements into recognizable wholes. Neural networks are used to carry out the minimization.

Previous work on optimization in vision (Barrow and Popplestone 1971; Burr 1983; Hummel and Zucker 1983; Terzopoulos 1986) has typically been restricted to computations occurring at a single of level of abstraction and/or involving a single model. For example, surface interpolation schemes, even when they include discontinuities (Terzopoulos 1986) do not include explicit models for physical objects whose surface characteristics determine the expected degree of smoothness. By contrast, heterogeneous and hierarchical model-bases often occur in non-optimization approaches to visual recognition (Hanson and Riseman 1986) including some which use neural networks (Ballard 1986). We attempt to obtain greater expressibility and efficiency by incorporating hierarchies of abstraction into the optimization paradigm.

2 Casting Model Matching as Optimization

We consider a type of objective function which, when minimized by a neural network, is capable of expressing many of the ideas found in frame systems in Artificial Intelligence (Minsky 1975). These "Frameville" objective functions (Mjolsness et al. 1988) are particularly well suited to applications in model-based vision, with frames acting as few-parameter abstractions of visual objects or perceptual groupings thereof. Each frame contains real-valued parameters, pointers to other frames, and pointers to predefined models (for example, models of objects in the world) which determine what portion of the objective function acts upon a given frame.

2.1 Model Matching as Graph Matching. Model matching involves finding a match between a set of frames, ultimately derived from visual data, and the predefined static models. A set of pointers represent object-part relationships between frames, and are encoded as a graph or sparse matrix called ina. That is, $ina_{ij} = 0$ unless frame j is "in" frame i as one of its parts, in which case $ina_{ij} = 1$ is a "pointer" from j to i. The expected object-part relationships between the corresponding models is encoded as a fixed graph or sparse matrix INA. A form of inexact graph-matching is required: ina should follow INA as much as is consistent with the data.

A sparse match matrix M ($0 \leq M_{\alpha i} \leq 1$) of dynamic variables represents the correspondence between model α and frame i. To find the best match between the two graphs one can minimize a simple objective function for this match matrix, due to Hopfield (1984) (also Feldman et

al. 1988; von der Malsburg and Bienenstock 1986), which just counts the number of consistent rectangles (see Fig. 1a):

$$E(M) = -\sum_{\alpha\beta}\sum_{ij} INA_{\alpha\beta} ina_{ij} M_{\alpha i} M_{\beta j}. \tag{2.1}$$

This expression may be understood as follows: For model α and frame i, the match value $M_{\alpha i}$ is to be increased if the neighbors of α (in the *INA* graph) match to the neighbors of i (in the *ina* graph).

Note that $E(M)$ as defined above can be trivially minimized by setting all the elements of the match matrix to unity. However, to do so will violate additional syntactic constraints of the form $h(M) = 0$ which are imposed on the optimization, either exactly (Platt and Barr 1988) or as penalty terms (Hopfield and Tank 1985) $\frac{c}{2}h^2(M)$ added to the objective function. Originally the syntactic constraints simply meant that each frame should match one model and vice versa, as in (Hopfield and Tank 1985). But in Frameville, a frame can match both a model and one of its specializations (described later), and a single model can match any number of instances or frames. In addition one can usually formulate constraints stating that if a model matches a frame then two distinct parts of the same model must match two distinct part frames and vice versa. We have found the following formulation to be useful:

$$\sum_{\alpha} INA_{\alpha\beta} M_{\alpha i} - \sum_{j} ina_{ij} M_{\beta j} = 0, \quad \forall \beta, i \tag{2.2}$$

$$\sum_{i} ina_{ij} M_{\alpha i} - \sum_{\beta} INA_{\alpha\beta} M_{\beta j} = 0, \quad \forall \alpha, j \tag{2.3}$$

where the first sum in each equation is necessary when several high-level models (or frames) share a part. (It turns out that the first sums can be forced to zero or one by other constraints.) The resulting competition is illustrated in figure 1b. Another constraint is that M should be binary-valued, i.e.,

$$M_{\alpha i}(1 - M_{\alpha i}) = 0, \tag{2.4}$$

but this constraint can also be handled by a special "analog gain" term in the objective function (Hopfield and Tank 1985) together with a penalty term $c\sum_{\alpha i} M_{\alpha i}(1 - M_{\alpha i})$.

In Frameville, the *ina* graph actually becomes variable, and is determined by a dynamic grouping or "perceptual organization" process. These new variables require new constraints, starting with $ina_{ij}(1-ina_{ij}) = 0$, and including many high-level constraints which we now formulate.

2.2 Frames and Objective Functions. Frames can be considered as bundles $\vec{F}_i$ of real-valued parameters F_{ip}, where p indexes the different parameters of a frame. For efficiency in computing complex arithmetic

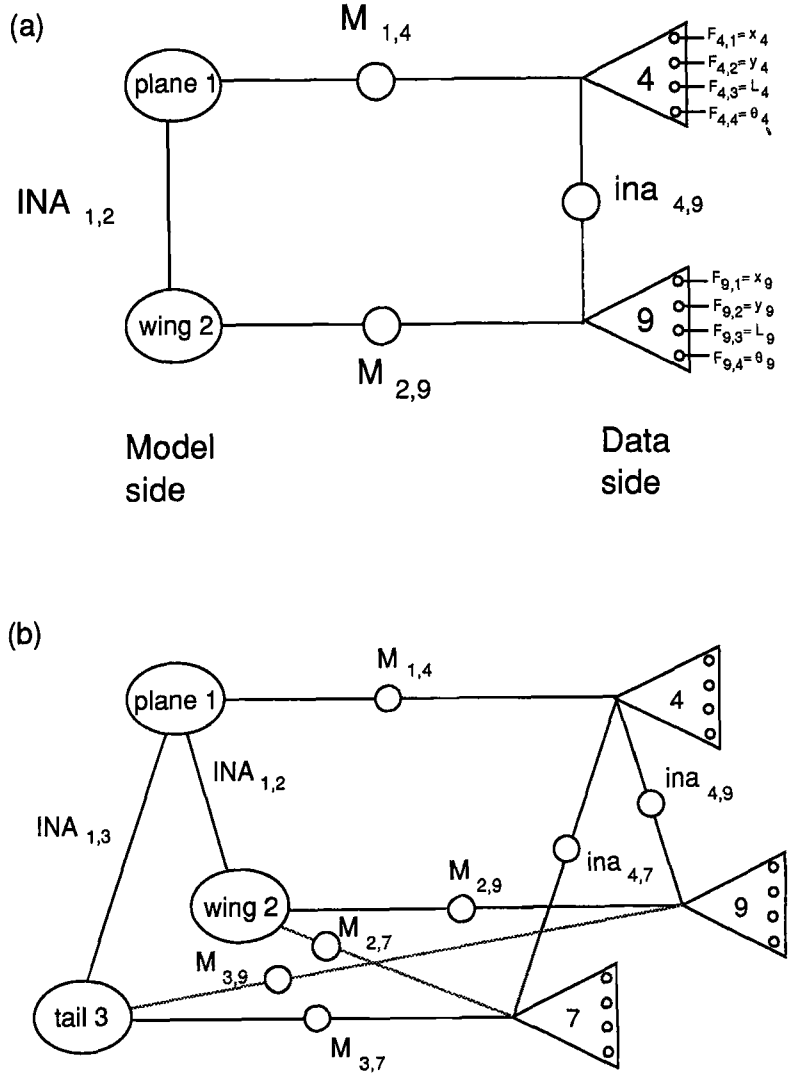

Figure 1: (a) Examples of Frameville rectangle rule. Shows the rectangle relationship between frames (triangles) representing a *wing* of a plane, and the *plane* itself. Circles denote dynamic variables, ovals denote models, and triangles denote frames. For the *plane* and *wing* models, the first few parameters of a frame are interpreted as position, length, and orientation. (b) Frameville sibling competition among parts. The match variables along the shaded lines ($M_{3,9}$ and $M_{2,7}$) are suppressed in favor of those along the solid lines ($M_{2,9}$ and $M_{3,7}$).

relationships, such as those involved in coordinate transformations, an analog representation of these parameters is used. A frame contains no information concerning its match criteria or control flow; instead, the match criteria are expressed as objective functions and the control flow is determined by the particular choice of a minimization technique.

In Figure 1a, in order for the rectangle $(1, 4, 9, 2)$ to be consistent, the parameters F_{4p} and F_{9p} should satisfy a criterion dictated by models 1 and 2, such as a restriction on the difference in angles appropriate for a mildly swept back wing. Such a constraint results in the addition of the following term to the objective function:

$$\sum_{i,j,\alpha,\beta} INA_{\alpha\beta} ina_{ij} M_{\alpha i} M_{\beta j} H^{\alpha\beta}(\vec{F}_i, \vec{F}_j) \tag{2.5}$$

where $H^{\alpha\beta}(\vec{F}_i, \vec{F}_j)$ measures the deviation of the parameters of the data frames from that demanded by the models. The term H can express coordinate transformation arithmetic (for example, $H^{\alpha\beta}(x_i, x_j) = 1/2[x_i - x_j - \Delta x_{\alpha\beta}]^2$), and its action on a frame $\vec{F}_i$ is selectively controlled or "gated" by M and ina variables. This is a fundamental extension of the distance metric paradigm in pattern recognition; because of the complexity of the visual world, we use an entire database of distance metrics $H^{\alpha\beta}$.

We index the models (and, indirectly, the database of H metrics) by introducing a static graph of pointers $ISA_{\alpha\beta}$ to act as both a specialization hierarchy and a discrimination network for visual recognition. A frame may simultaneously match to a model and just one of its specializations:

$$M_{\alpha i} - \sum_{\beta} ISA_{\alpha\beta} M_{\beta i} = 0. \tag{2.6}$$

As a result, ISA siblings compete for matches to a given frame (see Figure 2); this competition allows the network to act as a discrimination tree.

Frameville networks have great expressive power, but have a potentially serious problem with cost: for n data frames and m models there may be $O(nm+n^2)$ neurons widely interconnected but sparsely activated. The number of connections is at most the number of monomials in the polynomial objective function, namely n^2mf, where f is the fan-out of the INA graph. One solution to the cost problem, used in the line grouping experiments reported in section 3.2, is to restrict the flexibility of the frame system by setting most M and ina neurons to zero permanently. The few remaining variables can form an efficient data structure such as a pyramid in vision. A more flexible solution might enforce the sparseness constraints on the M and ina neurons during minimization, as well as at the fixed point. Then large savings could result from using "virtual" neurons (and connections) which are created and destroyed dynamically. This and other cost-cutting methods are a subject of continuing research.

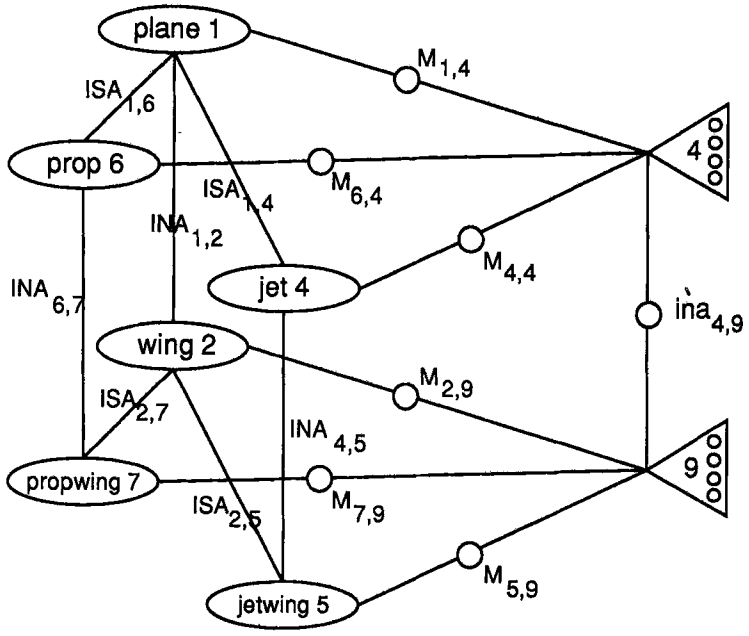

Figure 2: Frameville specialization hierarchy. The *plane* model specializes along *ISA* links to a *propeller plane* or a *jet plane* and correspondingly the *wing* model specializes to *prop-wing* or *jet-wing*. Sibling match variables $M_{6,4}$ and $M_{4,4}$ compete as do $M_{7,9}$ and $M_{5,9}$. The winner in these competitions is determined by the consistency of the appropriate rectangles, for example, if the 4-4-9-5 rectangle is more consistent than the 6-4-9-7 rectangle, then the *jet* model is favored over the *prop* model.

3 Experimental Results

3.1 Recognizing Simple Shapes. Frameville experiments were conducted in a domain consisting of a two-level compositional hierarchy. As

seen in Figure 3a, the input data at the lowest level are unit-length line segments parameterized by location x, y and orientation θ, corresponding to frame parameters F_{jp} ($p = 1, 2, 3$). We allow only horizontal ($\theta = 0$) and vertical ($\theta = \pi/2$) orientations. There are two high-level models, "T" and "L" junctions, each composed of three low-level segments. The task is to recognize instances of "T", "L", and their parts, in a translation-invariant manner. The high-level models are abstracted by the parameters of a designated main part, in this case, the upper vertical segment of each model.

On the model side, there are seven low-level models indexed by β, as shown in Figure 3b. These correspond to seven positional roles that a segment may assume in the context of a composite figure. These positions are illustrated iconically inside the model nodes in Figure 3b and correspond to the positions of segments in the familiar seven-segment LED display. The high-level models "T" and "L", indexed by α, are then specified by the appropriate set of *INA* links. We distinguish between high-level frames, indexed by i, that may match only high-level junction models, and low-level frames, indexed by j, that may match only low-level segment models.

For this domain, the parameter check term $H^{\alpha\beta}$ of Equation 2.5 checks the location and orientation of a given part relative to the main part. For example, in recognizing a "T", if low-level frame 3 is matched to model 5, a "middle horizontal" segment, then its parameters ($F_{3,1}$, $F_{3,2}$, $F_{3,3} = x_3, y_3, \theta_3$) must differ from those of an "upper vertical" mainpart by quantities $+1$, -1, and $\pi/2$, respectively. In our design the parameters of a high-level frame represent a best fit to the parameters of its mainpart. So if high-level frame 7 is matched to model 9, a "T", then an appropriate parameter check term is:

$$H^{9,5}(\vec{F}_7, \vec{F}_3) = (F_{7,1} - F_{3,1} - 1)^2 + (F_{7,2} - F_{3,2} + 1)^2 + (F_{7,3} - F_{3,3} - \frac{\pi}{2})^2. \quad (3.1)$$

The quantities $+1$, -1, and $\pi/2$ are thus model information stored in the objective function. This kind of objective function also determines the best-fit high-level parameters $\vec{F}_7$, even if the low-level mainpart frame itself is missing. Note here that a limited form of invariance is achieved by analog computation of relative coordinates; instances of "T" and "L" are recognized in a manner invariant to translation. (Rotation invariance can also be formulated if a different parameterization is used, but no experiments have been done.)

We used the unconstrained optimization technique in (Hopfield and Tank 1985) to minimize the objective function. We achieved improved results by including terms demanding that at most one model match a given frame, and that at most one high-level frame include a given low-level frame as its part. These are expressed as additive penalty terms:

$$\sum_{i\alpha} \sum_{\alpha' \neq \alpha} M_{\alpha i} M_{\alpha' i}, \quad \sum_{j\beta} \sum_{\beta' \neq \beta} M_{\beta j} M_{\beta' j}, \quad \sum_{ji} \sum_{i' \neq i} ina_{ij} ina_{i'j} \; . \quad (3.2)$$

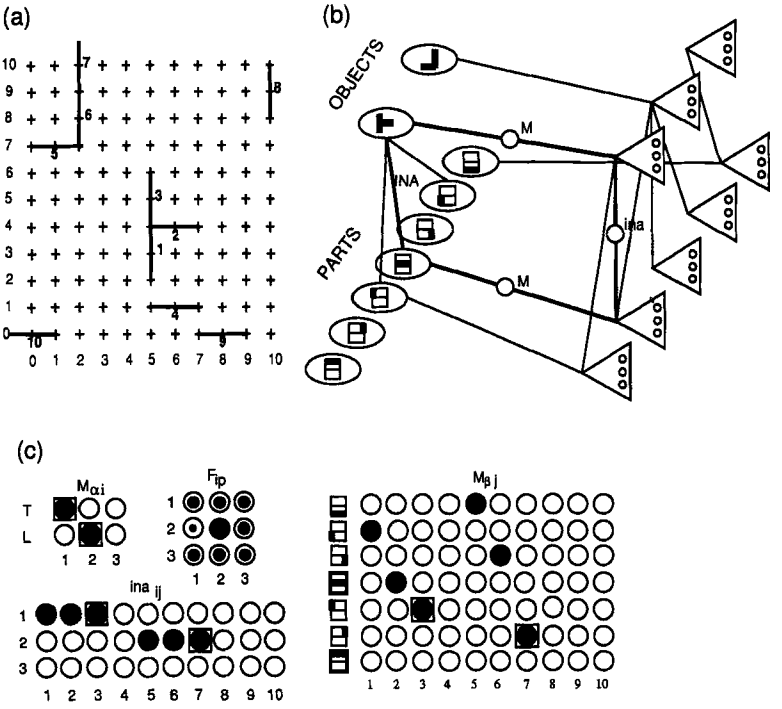

Figure 3: (a) Input data consists of unit-length segments oriented horizontally or vertically. The task is translation-invariant recognition of three segments forming a "T" junction (for example, sticks 1,2,3) or an "L" (for example, sticks 5, 6, 7) amid extraneous noise sticks. (b) Structure of network. Models occur at two levels. *INA* links are shown for a "T". Each frame has three parameters: position x, y and orientation θ. Also shown are some match and ina links. The bold lines highlight a possible consistency rectangle. (c) Experimental result. The value of each dynamical variable is displayed as the relative area of the shaded portion of a circle. Matrix $M_{\beta j}$ indicates low-level matches and $M_{\alpha i}$ indicates high-level matches. Grouping of low-level to high-level frames is indicated by the ina matrix. The parameters of the high-level frames are displayed in the matrix F_{ip} of linear analog neurons. (The parameters of the low-level frames, held fixed, are not displayed.) The few neurons circumscribed by a square, corresponding to correct matches for the main parts of each model, are clamped to a value near unity. Shaded circles indicate the final correct state.

In addition, we did not include the binary-value constraint (Equation 2.4). The linear analog neurons representing parameters in frames were not sigmoidally mapped as in (Hopfield and Tank 1985).

Figure 3c shows results of attempts to recognize the two junctions in Figure 3a. When initialized to small random values, the network becomes trapped in unfavorable local minima of the fifth-order objective function. (With only a *single* high-level model in the database, the system recognizes a shape amid noise given a random start.) If, however, the network is given a "hint" in the form of an initial state with mainparts and high-level matches set correctly, the network converges to the correct stable state. In particular, the linear parameter neurons settle to correct analog values corresponding to position and orientation of the mainparts of the junctions. Also, the proper dynamic grouping is accomplished as the ina neurons achieve the correct values, and the segment frames match the proper low-level models. Extraneous "noise" sticks remain unmatched.

There is a great deal of unexploited freedom in the design of the model base and its objective functions; there may be good design disciplines which avoid introducing spurious local minima. For example, it may be possible to use *ISA* and *INA* hierarchies to guide a network to the desired local minimum.

3.2 Line Grouping. Frameville is also being applied to the problem of extracting long straight lines from an image by recursively grouping smaller line segments into longer lines. The model base for the initial experiments shown in Figure 4a consists of lines at two levels, denoted 0 and 1. The level-1 line is composed of a left-line and a right-line at level 0, which are specializations of the level-0 line. We have conducted experiments on problems involving a 3×3 grid of level-0 frames and a 2×2 grid of level-1 frames. Each level-1 frame is connected to four level-0 frames that are near its spatial location, thus forming an overlapped pyramid. The end points of level-0 lines are specified as input data. Each level-1 line is denoted by four points, which correspond to the projections of the end points of its two component level-0 lines. The points on the level-1 line are determined by minimizing the energies of the springs shown in Figure 4b. The level-1 line frames contain additional slots that are used in the verification of colinearity of the four points on the line.

The results in Figure 4 were obtained by using the syntactic constraints of Equations 2.2 and 2.3 as penalty terms, while exactly maintaining the binary-value constraint of Equation 2.4 (both on the M and the ina variables). The constrained optimization method described in (Platt and Barr 1988) was used. The distance metric H in Equation 2.5 measures the energies of the springs shown in Figure 4b. To achieve stability to the network, Equation 2.5 was modified by replacing the M and ina variables by their squares. The details of our model-base and the constraints as

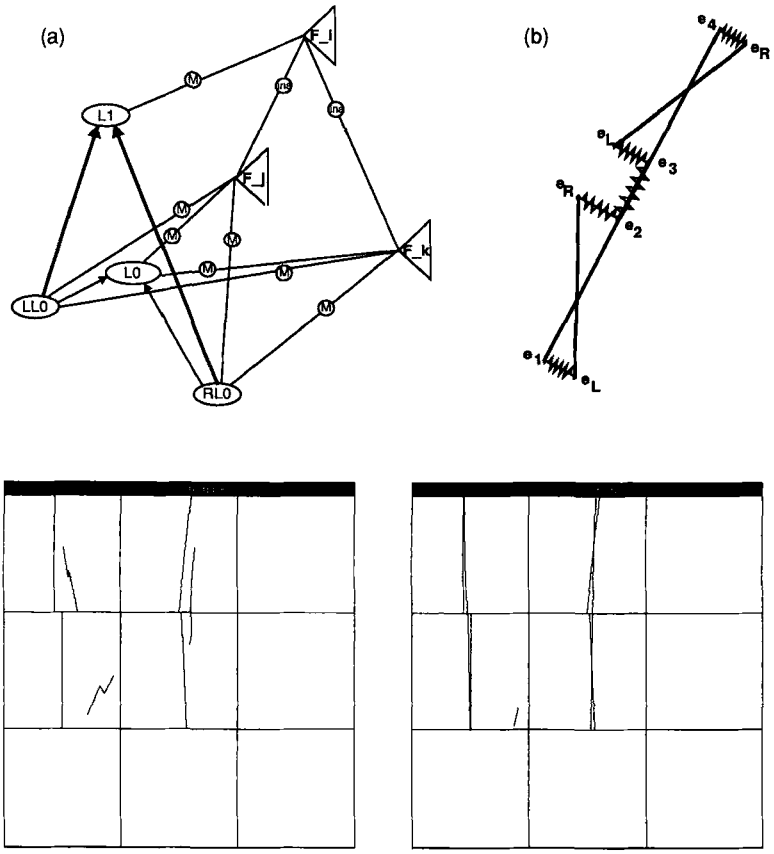

Figure 4: (a) The line model base. The thin lines connecting models represent *ISA* links and the thick lines represent the *INA* links. (b) The spring-stick model of fit. The springs between the level-0 lines and the level-1 line favor colinearity, while the spring between the two intermediate points on the longer line favors spatial proximity of the component lines. (c) and (d) Experimental results. The input data consists of the four vertical line segments and the approximating level-1 lines are displayed as three segments connecting the four points $\vec{e}_1 \ldots \vec{e}_4$ as seen in (b). In (c) the network is at an intermediate stage, where two of the segments have been correctly grouped, while the other two line segments appear to become parts of different high-level frames. In (d) the network has moved away from the incorrect solution and is close to the correct solution. The small extra line segment eventually vanishes.

well as planned extensions of this work are described in (Anandan et al. 1989).

4 Conclusion

Frameville provides opportunities for integrating all levels of vision in a uniform notation which yields analog neural networks. Low-level models such as fixed convolution filters just require analog arithmetic for frame parameters, which is provided. High-level vision typically requires structural matching, also provided. Qualitatively different models may be integrated by specifying their interactions, $H^{\alpha\beta}$.

Acknowledgments

We wish to thank Joachim Utans, John Ockerbloom, and Charles Garrett for the Frameville simulations. This work was supported in part by AFOSR grant F49620-88-C-0025, by DARPA grant DAAA15-87-K-0001, and by ONR grant N00014-86-0310.

References

Anandan, P., E. Mjolsness, and G. Gindi. 1989. *Low-level visual grouping via optimization in neural networks.* Technical Report, Yale University Computer Science Department. Manuscript in preparation.

Ballard, D. 1986. Cortical connections and parallel processing: Structure and function. *Behavioral and Brain Sciences*, **9**, 67–120.

Barrow, H.G., and R.J. Popplestone. 1971. Relational descriptions in picture processing. *In:* Machine Intelligence, **6**, ed. D. Mitchie. Edinburgh: University Press.

Burr, D.J. 1983. Matching elastic templates. *In:* Proceedings of the International Symposium on Physical and Biological Processing of Images, eds. O.J. Braddick and A.C. Sleigh. Springer-Verlag.

Feldman, J.A., M.A. Fanty, and N.H. Goddard. 1988. Computing with structured neural networks. *IEEE Computer*, **21:3**, 91–103.

Hopfield, J.J. 1984. Personal communication.

Hopfield, J.J. and D.W. Tank. 1985. Neural computation of decisions in optimization problems. *Biological Cybernetics*, **52**, 141–152.

Hummel, R.A. and S.W. Zucker. 1983. On the foundations of relaxation labeling processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-5**, 267–287.

Minsky, M.L. 1975. A framework for representing knowledge. *In:* The Psychology of Computer Vision, ed. P.H. Winston, 211-277. McGraw-Hill.

Mjolsness, E., G. Gindi, and P. Anandan. 1988. *Optimization in model matching and perceptual organization: A first look.* Technical Report YALEU/DCS/RR-634, Yale University.

Platt, J.C. and A.H. Barr. 1988. Constraint methods for flexible models. *Computer Graphics*, **22:4**, 279–288. Proceedings of SIGGRAPH '88.

Riseman, E.M. and A.R. Hanson. 1986. A methodology for the development of general knowledge-based vision systems. *In:* Vision, Brain, and Cooperative Computation, eds. M.A. Arbib and A.R. Hanson, 285–328. MIT Press.

Terzopoulos, D. 1986. Regularization of inverse problems involving discontinuities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-8**, 413–424.

von der Malsburg, C. and E. Bienenstock. 1986. Statistical coding and short-term synaptic plasticity: A scheme for knowledge representation in the brain. *In:* Disordered Systems and Biological Organization, 247–252. Springer-Verlag.

Received 1 October 1988; accepted 17 October 1988.