Learning Recognition and Segmentation of 3-D Objects from 2-D Images

John J. Weng

Department of Computer Science
Michigan State University
East Lansing, MI 48824 USA

N. Ahuja and T. S. Huang

Beckman Institute
University of Illinois
Urbana, IL 61801 USA

Abstract

A framework called Cresceptron is introduced for automatic algorithm design through learning of concepts and rules, thus deviating from the traditional mode in which humans specify the rules comprising a vision algorithm. With the Cresceptron, humans as designers need only to provide a good structure for learning, but they are relieved of most design details. The Cresceptron has been tested on the task of visual recognition: recognizing 3-D general objects from 2-D photographic images of natural scenes and segmenting the recognized objects from the cluttered image background. The Cresceptron uses a hierarchical structure to grow networks automatically, adaptively and incrementally through learning. The Cresceptron makes it possible to generalize training exemplars to other perceptually equivalent items. Experiments with a variety of real-world images are reported to demonstrate the feasibility of learning in the Cresceptron.

1 Introduction

In real-world vision problems, the factors that affect the intensity of an image change constantly. Most of these factors are unknown and uncontrollable in a general setting, making computer vision difficult and the progress in computer vision slow [9].

1.1 Approaches to Vision

Manually developing vision rules. Currently prevailing approaches to computer vision rely on human designers to manually develop a set of rules for a specific task and then to explicitly code these rules into a program. In order to make the problem manually tractable, many assumptions are made. The systems constructed using this type of approaches tend to be brittle: they fail in situations where one or more of the assumptions are not satisfied. Moreover, this approach is not scalable: it is intractable to manually design a set of rules that are sufficient to deal with complex vision problems in the real world.

Learning in human vision. In contrast, human vision seems extremely versatile. It is known that learning takes place over a long period and plays a central role in the development of such a capability in humans. Human vision appears to be more a process of learning and recalling rather than one relying on understanding the physical processes of image formation and object-modeling. As demonstrated by the "Thatcher's illusion" [13], facial expression is very difficult to recognize from an upside-down face, although it would be quickly revealed by a simple "mental" rotation if the brain could perform such a rotation. The evidence for learning in vision includes even low-level vision. For instance, a common visual experience, overhead light source, is learned and used to perceive shape from shading [12], although the solution to the problem is not unique from the image formation point of view.

Learning in computer vision. For complex vision problems, self-organizing through self-learning is a promising approach. The idea of learning for vision is not new. It is the message that comes through most clearly from the work in psychology, cognitive science and neurophysiology [4] [1] [12] [7]. The question is how to do computational learning.

1.2 Learning Techniques

Many decision-making problems fall into the general category of classification. The classification methods can be roughly divided into two types: statistical pattern recognition methods and symbolic methods. Recently, there has been a surge in interest in learning using models of artificial neural networks (or connectionist models of computation) [14] [10] [6].

Most studies on learning assume that a feature-vector description of objects is available, presumably

0-8186-3870-2/93 $3.00 © 1993 IEEE

extracted by humans. However extraction of objects from images and computation of their descriptions is a major task. If a human is available to segment the objects of interest from images, then why not let her/him do the entire recognition! If feature vectors are provided for the entire image without identifying which features belong to a single object, no traditional learning technique will work.

Some studies of learning from retinotopic data (i.e., each data item corresponds to a sensory position on the retina) can be found in the literature. The Neocognitron by Fukushima and his colleagues [2] [3] was designed for recognizing a small number of segmented patterns such as numerals and alphabets. Pomerleau's work [11] demonstrated that the performance of a neural-network-controlled CMU NAVLAB in road following is comparable to that achieved by the best traditional vision-based autonomous navigation algorithm at CMU.

1.3 The Challenges

Although the use of neural networks has shown encouraging results, it is not clear whether this approach can deal with complex real-world recognition-and-segmentation problems for which a retinotopic network is needed. There is a lack of systematic treatment of the retinotopic network structure, and the theory for such neural networks is missing. Neural network is most treated as an opaque box and its learning is often formulated as an optimization problem with a huge number of parameters. A consequence of this situation is the unpredictable performance of the network. Sometimes, a backpropagation learning algorithm leads to a good network but often it does not. To handle the complexity of general vision problems, we have identified the following requirements:

- The system must be able to automatically learn the rules that human (practically) cannot manually specify. Learning should not be limited to the parameters of manually selected rules, because a fixed set of rules is not scalable to complex problems.
- Knowledge representation must be automatic: it is intractable to manually define the feature represented by every neuron. Significant image structures, or concepts, must be automatically identified, and their breakdown and mapping to the framework must be automatic.
- Learning must be reliable. The unpredicatable performance as with backpropagation learning must be avoided.

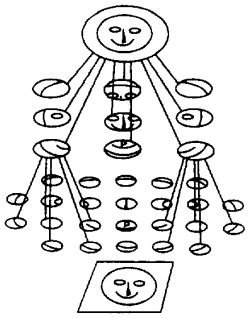


Figure 1: A schematic illustration of hierarchical feature grouping in the Cresceptron. In the figure, not all the connections are shown.

- Learning must be fast: the size of a network for a complex vision task has to be large. Repeated modification of all weights is impractical.
- Learning must be incremental: an addition of a new object to be learned should not require the entire network to be re-trained. This is a key towards a self-improving complex vision system.

1.4 The Cresceptron

Our framework is called *Cresceptron*, coined from Latin *cresco* (grow) and *perceptio* (perception). Like Neocognitron, this framework uses a multi-level retinotopic layers of neurons. However, it is fundamentally different from the Neocognitron in that, among other things, the network configuration of the Cresceptron is ***automatically*** determined during learning. The following are some salient features of the Cresceptron which contribute to the satisfaction of the above mentioned requirements.

1. The Cresceptron uses unsupervised learning from automatic hierarchical image ***analysis*** and hierarchical structural concepts derived therefrom (see Fig. 1). Learning in the Cresceptron is incremental and thus growth is possible. New concepts are detected and the network components are appropriately created to relate new concepts with previously learned concepts. Knowledge sharing occurs automatically at every level of the network which keeps the network size limited.
2. Tolerance to deviation is made hierarchical, smaller at lower levels and larger at high levels. This makes it possible to handle many perceptually similar objects from a relatively small set of training examplars.
3. Learning is based on hierarchical ***analysis*** instead of back-propagation. Therefore, the network is

not an opaque box, and the problem of local minima with the back-propagation methods is avoided.

4. Segmentation and recognition are simultaneous. No foreground extraction is necessary.
5. The network is locally connected, not globally, for efficiency.

2 The Network Components

The Cresceptron network consists of several levels (7 in the current version). The number of levels is to guarantee that the receptive field of each top-level node covers the entire fovea image.

Each level has 2 or 3 layers. Thus, totally, the network has several layers that are numbered by l, $l = 0, 1, 2, ..., L$. The output of a lower layer l is the input for the next higher layer $l + 1$. At each layer l, there are many neural planes. Each neural plane consists of a square of $k(l) \times k(l)$ nodes. Since each neural plane represents a concept and the response at a certain location of the neural plane indicates the presence of the concept, all the locations in a neural plane share the same sigmoidal function and the same set of synaptic weights. The receptive field of a node at a layer l is defined as the spatial extent of the layer-0 input pixels it connects to either directly, or indirectly through other intermediate lower layers. We first briefly describe basic components of the Cresceptron network.

2.1 Pattern-Detection Layer

The purpose of the pattern-detection layer is to detect the presence of a feature at all locations. Two types of pattern-detection layer are useful: regular pattern-detection layer and subsampled pattern-detection layer.

The regular pattern-detection layer is illustrated in Fig. 2. Let $n(l, m, i, j)$ denote the value of response at position (i, j) in the m-th neural plane at layer l. A concept at position (i_0, j_0) is a 2-D pattern $\{n(l, m, i_0 + i, j_0 + j) \mid -h \leq i, j \leq h\}$. In the learning phase, once a new concept is detected at (i_0, j_0) at layer l, a new neural plane k is created at layer $l + 1$ which is devoted to this concept. The new concept is memorized by a new node whose synapses are assigned with the observed values

$$w(l, k, m, i, j) = n(l, m, i_0 + i, j_0 + j),$$

$-h \leq i, j \leq h$. Let P denote the set of all the indices of input planes where the new concept is detected. In the recognition phase, the response in the k-th new neural plane at (i_0, j_0) of layer $l + 1$ is

$$n(l+1, k, i_0, j_0) = f[s(l+1, k)z(l, k, i_0, j_0) - T(l+1, k)]$$

where

$$z(l, k, i_0, j_0) = \sum_{m \in P} \sum_{-h \leq i,j \leq h} w(l, k, m, i, j) n(l, m, i_0 + i, j_0 + j)$$

and $f(x)$ is a sigmoidal function [6] that maps x to a normalized range $[0, 1]$, and the values $s(l + 1, k)$ and $T(l + 1, k)$ are automatically determined in the learning phase so that

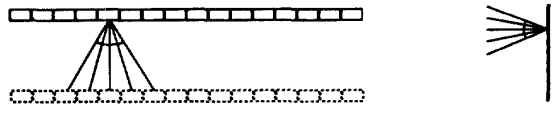
$$f[v\, s(l + 1, k)z(l, k, i_0, j_0) - T(l + 1, k)] \approx 1 \quad (1)$$

and

$$f[\frac{v}{2} s(l + 1, k)z(l, k, i_0, j_0) - T(l + 1, k)] \approx 0 \quad (2)$$

where v is a user-specified *system vigilance* parameter.

Another type, the subsampled pattern-detection layer, is similar to the regular one except that the input nodes are subsamples of the input neuron array with a subsample spacing r (one sample every r nodes) as illustrated in Fig. 3. In our system, r is such that the receptive fields of these four subsamples have a minimal overlap.



(a) (b)

Figure 2: Regular pattern-detection layer. (a) A schematic illustration in which only the connections to one node are drawn and only one input plane is shown. The arc across the connections represents an AND-like condition. (b) The symbol of the regular pattern-detection layer. The number of connections indicates the size value $2h + 1$. But, the case of 2 connections is reserved for the subsampled pattern-detection layer.

2.2 Node-Reduction Layer

If the number of nodes in each neural plane is reduced from layer l to $l + 1$, then we say that $l + 1$ is a *node-reduction layer* which is shown in Fig. 4. We use node-reduction layer to increase the space connectivity, and reduce the spatial resolution. At a node

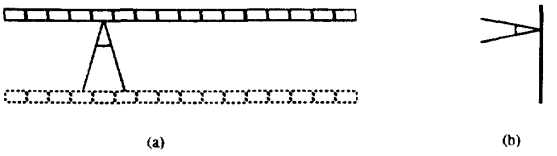


Figure 3: Subsampled pattern-detection layer. (a) A schematic illustration. The arc across the connections represents an AND-like condition. (b) The symbol of the subsampled pattern-detection layer.

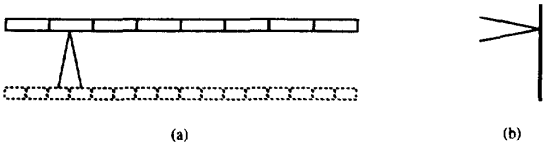


Figure 4: Node reduction layer. (a) A schematic illustration. No arc across the connections is present, which represents an OR-like condition. (b) The symbol of the subsampled pattern-detection layer.

reduction layer, the value of a node-reduction neural plane k that accepts the input from neural plane m at layer l is determined by:

$$n(l+1, k, i, j) = \max\{n(l, m, 2i+p, 2j+q) \mid p, q = 0, 1\}.$$

Definition 1 *If a network learns, in the learning phase, a pattern that is presented at a certain location in the input, and it can also recognize, in the later recognition phase, the same pattern but translated arbitrarily in the input image, then, this network is* **recallable under translation**.

2.3 Node-Reduction Modules

Definition 2 *A* **grid-centered node-reduction module** *(GCNR module) consists of two layers: the lower layer is a regular pattern-detection layer and the upper layer is a node-reduction layer. The output of the lower layer is the input of the upper layer.*

Property 1 *The GCNR module is recallable under translation.*

2.4 Blurring Layer

Suppose layer $l+1$ is a blurring layer. Let $n(l, i_0, j_0)$ denote the response at position (i_0, j_0) in a neural plane at input layer l. Then the output at position (i_0, j_0) of the corresponding neural plane at layer $l+1$

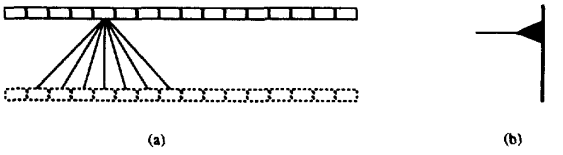


Figure 5: Blurring layer. (a) A schematic illustration in which only the connections to one node are shown. Every neural plane at the blurring layer has only one input plane. No arc across the connections is present, which represents an OR-like condition. (b) The symbol of the blurring layer. The black triangle represents the contribution from a single input node.

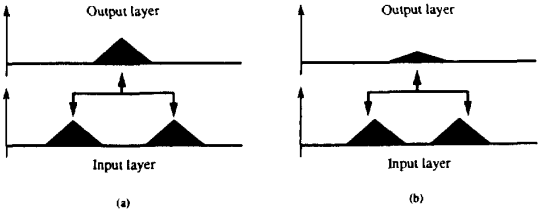


Figure 6: The mechanism of detection and measurement of geometric configuration of features from input layers. In the input layer, the position of a feature is represented by a peak. The bluring of the peak enables the output layer to measure the positional accuracy. (a) If the positions are exactly correct, two peaks are sensed and thus, the response is high at the output layer. (b) If the positions are displaced, the slopes are sensed and thus, the output response is relatively low.

is defined by

$$n(l+1, i_0, j_0) = \max_{r \le R} \left\{ \frac{R-r}{R} n(l, i_0+i, j_0+j) \middle| r^2 = i^2 + j^2 \right\} \quad (3)$$

as illustrated in Fig. 5, where R is the radius of blurring, whose value depends on the receptive field. The blurring layer is designed to tolerate positional deviation, as shown in Fig. 6.

2.5 Hierarchical Networks

At each level, the amount of blurring should be proportional to the receptive field of the node, as shown by the example in Fig. 7. The framework for our network is illustrated in Fig. 8.

3 The Cresceptron

Visual attention. Visual attention defines a square attention window. The objective of visual attention

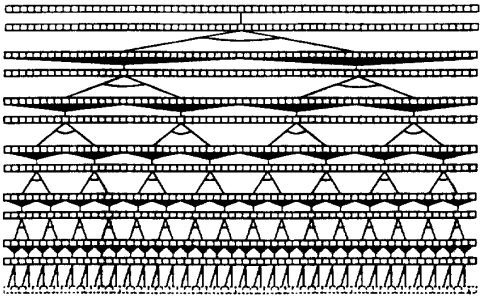


Figure 7: A 1-D illustration of a hierarchical network which consists of NCB modules (pattern-detection layer plus blurring layer). Note how the subsample spacing r and the amount of blurring change from low levels to high levels.

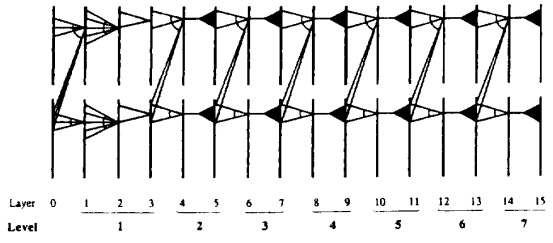


Figure 8: Schematic illustration of the selected framework for a multi-level network. In the illustration, a plane being connected to two lower-layer planes means that every plane in this layer can be connected to several planes during automatic learning. Otherwise, it accepts input from only one lower-layer plane.

is to scale the part of image covered by the square attention window down to the size of the "fovea" of the neural network. In our experiment, the fovea image is a square of 64×64 pixels.

In order to deal with object of different sizes, a series of *legal* attention window sizes are defined: $W_1, W_2, ..., W_k$, where $W_{j+1} = \alpha W_j$. (In our experiment $\alpha = 4/5$). There are two attention modes, manual and automatic. In the manual attention mode, which is mainly designed for the learning phase, the user interactively selects a location and a legal size of the attention window so that the object to be recognized can be directly mapped to the fovea. In the automatic attention mode, which is designed for the recognition phase, the system automatically scans the entire image, from a large attention window to small, with a step size (1/5 of attention window size). After a fovea image is obtained, learning or recognition is

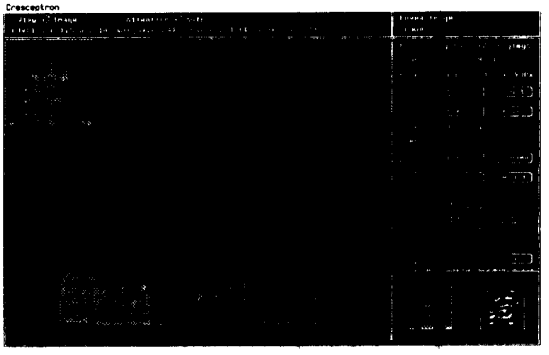


Figure 9: Interface console of the Cresceptron.

applied to it.

Image primitives. The system is designed in such a way that any image primitive can be easily used for learning and recognition. The current version of the Cresceptron uses directional edges as image primitives (zero-crossings of the second directional derivative of the Gaussian smoothed image along one of the 8 discretized directions, two different scales) as shown in the bottom two rows of Fig. 11.

Learning: what to learn. We have developed a window-based interactive interface shown in Fig. 9. During the learning phase, the user selects the object to learn by interactively draw a polygon in the fovea image to outline the object using a computer mouse.

Learning: detection of new concepts. New-concept detection is performed starting at layer 1, the pattern detection layer, and all the subsampled pattern-detection layers 4, 6, 8, 10, 12. An active pattern is significant if the response is high in the pattern. A new concept at (i_0, j_0) consists of the significant response at the location (i_0, j_0) in all the input neural planes. The concept is new if it has not been observed at any position at this level.

Learning: growth. Initially, the network does not exist: no neural plane or neurons exist at any layer. Given each input image to learn, the system automatically grows the network recursively starting from the lowest layer to the top layer. At each level, once a significant new concept is detected at position (i_0, j_0) in some lower-level neural planes, a new neuron with the synaptic connections is created together with a new neural plane that is devoted to this new concept. Finally, at the top layer, if the examplar is not recognized, a new plane is created at the top layer with a default label. The user assigns a meaningful name of the object to the label. Later in the recognition phase

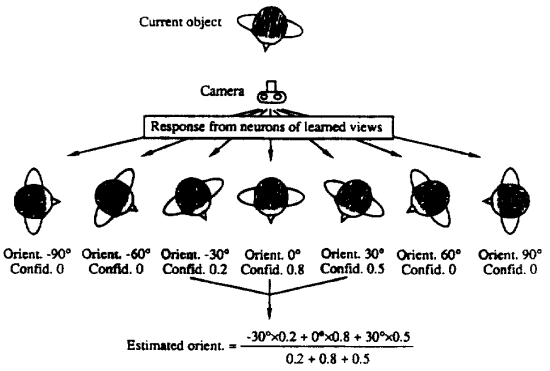


Figure 10: Recognize different orientations by learning several exemplars.

if this new neural plane is active at a position (i, j), then the label reports the name of the object being recognized at this position. The system can also learn several examplars in a class. To do this, the user identifies the top neural plane that represents this class and then clicks the button "class" instead of "learn". Thus, the system will share the same neural plane at the top layer.

Recognition and decision making. Fig. 11 shows the response of a few neural planes after recognition. At the top layer, the network reports all the response values (confidence values) higher than 0.5. When two or more different objects are reported, the one with the highest confidence is the one recognized. If multiple reports belong to the same type of object, further inference can be performed, as shown in Fig. 10.

Other variations. In the current version of the Cresceptron, scale and positional variations of the objects are addressed by visual attention coupled with the size tolerance and recallability under translation in the fovea recognition. A limited 3-D orientational tolerance is also obtained by the shape tolerance of the fovea recognizer. Significant variations should be learned individually. Some studies have demonstrated that the human vision system does not have a perfect invariance in either translation [8], scale [5], or orientation [13].

Segmentation. Once an object is recognized, the network can identify the location of the recognized object in the image. This is done by back tracking the response paths of the network from top layer down to the lowest layer.

4 Experiments

For the theoretical and algorithmic development, the Cresceptron system has been simulated on a SUN SPARC workstation with an interactive user interface to allow effortless training and examination of the network, as shown in Fig. 9.

Here we show the result from an automatically generated network that has been fully tested. This network has learned 21 classes of objects: 10 faces of different persons to test the power of discrimination, and 11 other objects to test the versatility, including a path scene, street car, dog, fire hydrant, walking human body, stop sign, parked car, telephone set, chair, and computer. This neural network was automatically created through learning of these objects.

The **tolerance** is demonstrated by correctly recognizing all the 35 expression images of a female reporter, extracted from a TV interview, based on a learning of three images. Some of these images are shown in Figure 12. The **discrimination power** is shown by correctly recognizing all the 10 faces learned, without confusion between different faces. Some of the faces are shown in Figs. 13 and 14. The **versatility** is displayed by successfully learned and recognized 11 different objects mentioned above. (See some in Figs. 13 and 14). The edge segments marked by the segmentation are shown in Fig. 14 for some objects recognized. It is interesting to observe that the edge segments are not completely connected and some are missing. The system does not rely on the connectivity of the detected edges nor on a close outline of the object.

After learning 25 objects, the network has a total of 4133 neural planes in the output layers of all the levels.

5 Conclusions

The result demonstrated a way of automatically creating an complex vision system by letting it learn, grow and organize by itself.

References

[1] J. R. Anderson, *Cognitive Psychology and Its Implications*, 3rd ed., Freeman, New York, 1990.

[2] K. Fukushima, "Cognitron: A self-organizing multilayered neural network," *Biological Cybernetics*, vol. 20, 1975, pp. 121-136.

[3] K. Fukushima, S. Miyake, and T. Ito, "Neocognitron: A neural network model for a mechanism of visual pattern recognition," *IEEE Trans. Sys-*

tems, Man, Cybernetics, vol. 13, no. 5, pp. 826-834, 1983.

[4] D. H. Hubel, *Eye, Brain, and Vision*, Scientific American Library, Vol. 22, 1988.

[5] P. A. Kolers, R. L. Duchnicky, and G. Sundstroem, "Size in visual processing of faces and words," *J. Exp. Psychol. Human Percept. Perform.*, vol. 11, pp. 726-751, 1985.

[6] R. P. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Magazine*, vol. 4, no. 2, April 1987, pp. 4-22.

[7] J. L. Martinez, Jr. and R. P. Kessner (eds.), *Learning & Memory: A Biological View*, 2nd ed., Academic Press, San Diego, 1991.

[8] T. A. Nazir and J. K. O'Regan, "Some results on translation invariance in the human visual system," *Spatial Vision*, vol. 5, no. 2, pp. 81-100, 1990.

[9] T. Pavlidis, "Why progress in machine vision is so slow," *Pattern Recognition Letters*, vol. 13, April 1992, pp. 221-225.

[10] T. Poggio and S. Edelman, "A network that learns to recognize three-dimensional objects," *Nature*, vol. 343, 1990, pp. 263-266.

[11] D. A. Pomerleau, "ALVINN: An autonomous Land Vehicle in a Neural Network," in D. Touretzky (ed), *Advances in Neural Information Processing*, vol. 1, Morgran-Kaufmann Publishers, San Mateo, CA, pp. 305-313.

[12] V. S. Ramachandran, "Perceiving shape from shading," in I. Rock (ed.), *The Perceptual World*, Freeman, San Francisco, CA, 1990, pp. 127-138.

[13] P. Thompson, "Margaret Thatcher: a new illusion," *Perception*, vol. 9, pp. 483-484, 1980.

[14] V. Vemuri (ed.), *Artificial Neural Networks: Theoretical Concepts*, IEEE Computer Society Press, Washington DC, 1988.

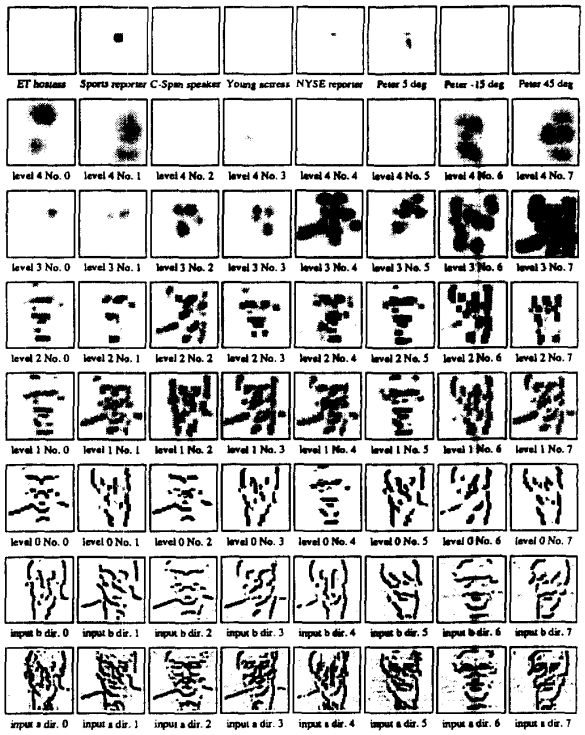


Figure 11: Response of the Cresceptron and inputs. The bottom two rows show the input edge images. The bottom row shows 8 directional edge images at a smaller scale and the row above it shows those with a larger scale. The rows 1 to 5 from top show the first several neural planes in the output layers of the levels 6, 4, 3, 2, 1, respectively.

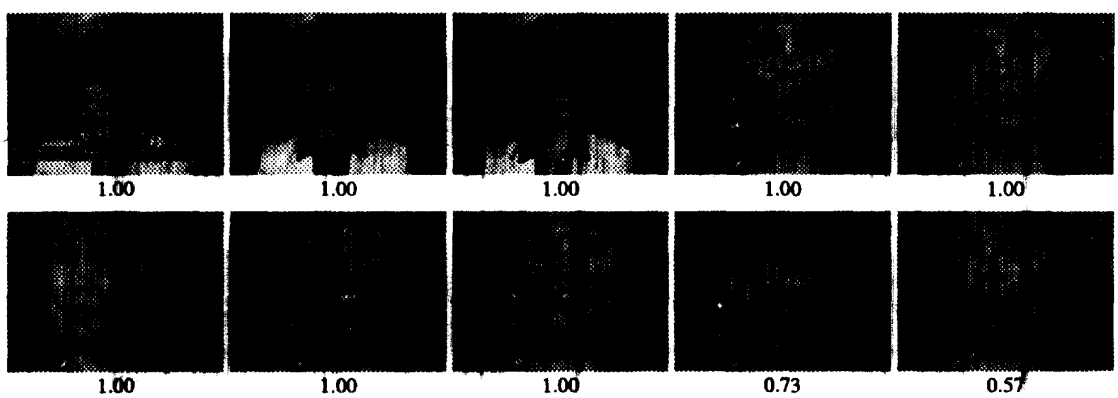


Figure 12: Face expressions in the images recognized as "ET Hostess" by the Cresceptron. The first three images are used to train the network. The number under each image is the response value (or confidence value) of the recognition, i.e., the response value at the corresponding node at the top layer.

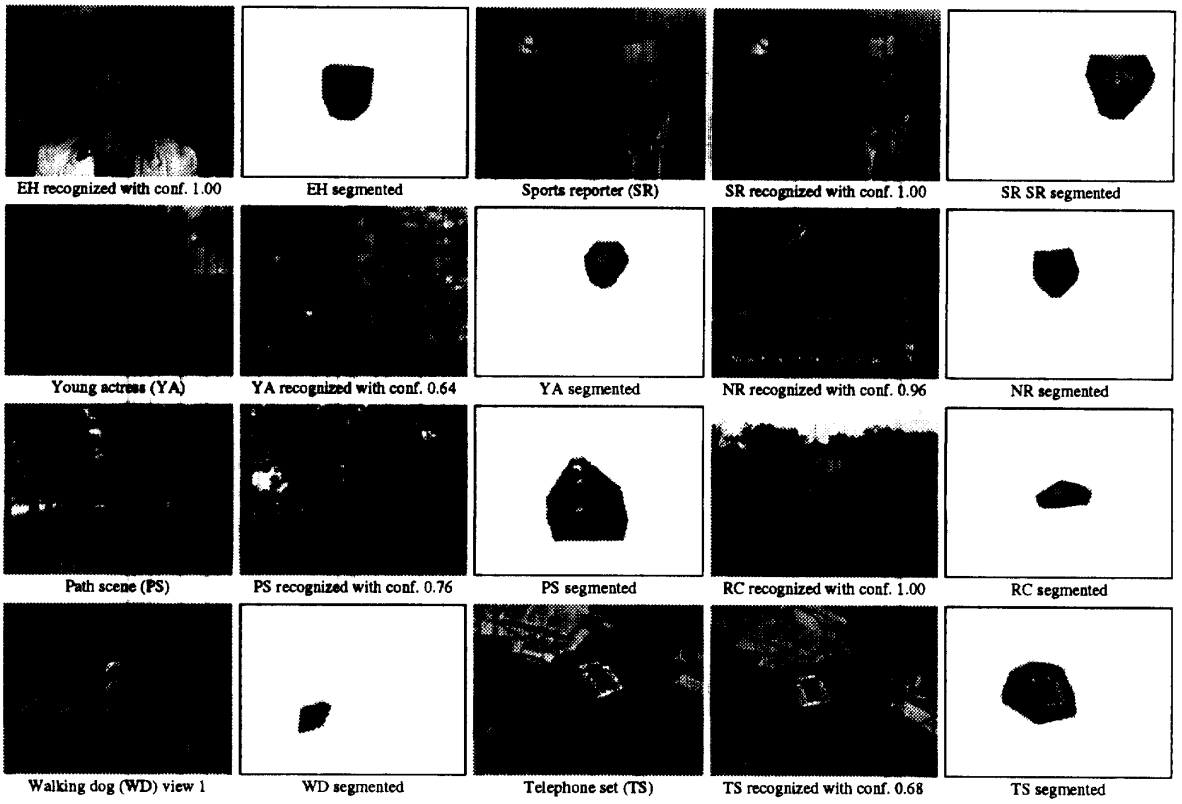


Figure 13: Some examples of learning, recognition and segmentation.

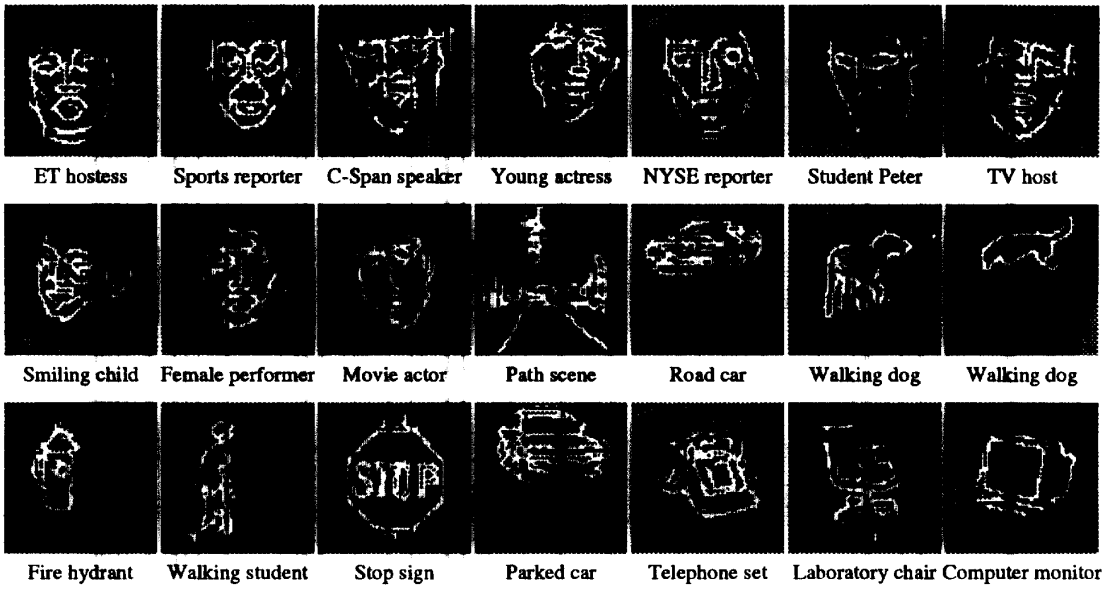


Figure 14: Edge segments marked by the segmentation process for some of the examples. These are the major edge segments contributing to the recognition.