INTEGRATION OF TRADITIONAL IMAGING, EXPERT SYSTEMS, AND NEURAL NETWORK TECHNIQUES FOR ENHANCED RECOGNITION OF HANDWRITTEN INFORMATION

Amar Gupta
John Riordan
Evelyn Roman

August 1990 WP # 3161-90

Integration of Traditional Imaging, Expert Systems, and Neural Network Techniques for Enhanced Recognition of Handwritten Information

Evelyn Roman, Amar Gupta, and John Riordan

 The research described herein has been supported by I.F.S.R.C. at M.I.T. All comments and suggestions should be directed to the Principal Investigator, Dr. Amar Gupta, Room E40-265, MIT, Cambridge, MA 02139; telephone (617) 253-8906.

ABSTRACT

This paper examines the interrelationship between traditional imaging techniques, expert systems, and neural networks. It begins with a discussion of neural network concepts. Then, a new taxonomy based on the quality and the complexity of a document is presented. Next, this paper analyzes the impact of neural networks in the context of reading handwritten material, using several examples. Finally, an architecture that utilizes contemporary neural network and expert system techniques, in conjunction with classical image processing, statistical and syntactic techniques, is described.

Keywords: Neural networks, expert systems, handwritten material, recognition, syntactic approaches, structural approaches, courtesy amounts, automated reading.

1 INTRODUCTION

The progression from printed material to unconstrained handwritten information is accompanied by a concurrent increase in noise, entropy, and difficulty of segmentation. In this paper, it is shown that increased complexity can be overcome by utilizing technology, which complements, not displaces, traditional imaging and expert system techniques. By optimal integration of image processing, statistical and syntactic pattern recognition, and expert systems techniques with newer neural network methodologies, one can increase the ability and the accuracy for reading information without human intervention.

Recognition of handwritten information is pertinent in many engineering, manufacturing, maintenance, and business applications. It has been researched in the context of reading postal zip codes in addresses on letters[3]. We opted to concentrate on the problem of reading the amount, expressed in numerical format, on checks. Fifty billion checks are processed each year in the United States alone [28]. The amount written in figures is called the courtesy amount; it consists of two portions – the dollar component and the cents component. There are many styles for writing the courtesy amount. While the account number on each check is printed in machine-readable typefonts, the courtesy amount must be read by a human operator and typed in. Systems involving multiple operators or batches must be utilized to ensure appropriate levels of accuracy. If the amount field could be read automatically, there is potential for reducing significant time and costs, and enhancing productivity. So far, imaging techniques have been largely utilized for storing images of checks and other documents in a bitmapped mode on optical disks[28]. If such information could be processed automatically, it would represent the next major step in automation.

Automated reading of handwritten material is not a trivial problem because of infinite variations of shapes caused by writing habits, styles, and even the moods of the writer[38]. Fortunately, in the case of checks, one must deal with numerals only, and the location of the field on a check is known in advance, unlike the postal zip code on a letter. However, while a certain error rate may be acceptable for reading zip codes, such a rate may be unacceptable in a banking environment. Further, the use of 12 different styles for writing the courtesy amount adds to the complexity of the problem. Conventional techniques for scanning cannot offer the desired accuracy level. Further, it is very difficult to formulate rules for all the diverse styles of writing courtesy amounts for various combinations of numerals. As such, one turns to neural network techniques, which attempt to perform functions that have so far been deemed to be beyond the scope of machines.

This paper first discusses fundamental neural network concepts. It then focuses on traditional image processing techniques, including methods for reducing noise and finding features. A new taxonomy for classifying documents is presented. The next section presents examples of design approaches adopted by researchers who have combined traditional approaches with new ones. This discussion is used to develop an architecture that utilizes contemporary neural network and expert system techniques, in conjunction with classical statistical and syntactic techniques, to provide higher ability to read handwritten material, along with increased accuracy and faster speed.

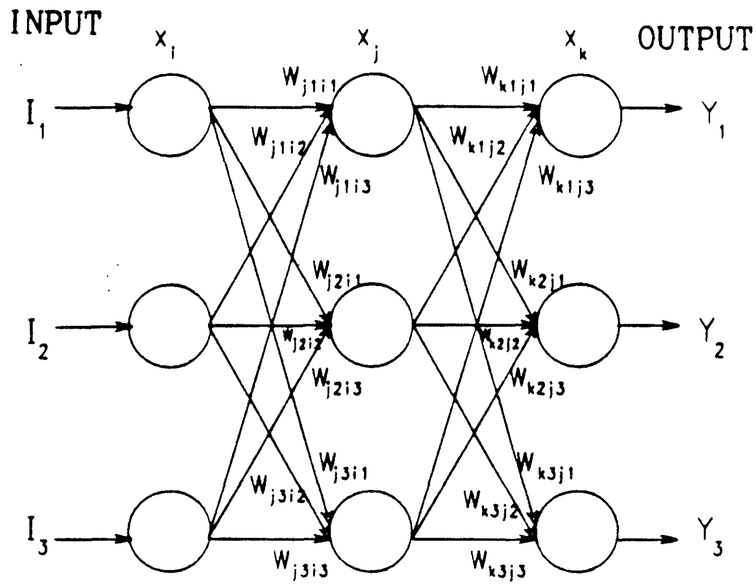


Figure 1: A fully connected three-layer neural network

2 Neural Networks

Artificial neural network technology encodes associations between features or concepts in terms of relative strengths of synaptic connections between neurons. Learning in neural networks involves the convergence of a set of function parameters (weights) to values enabling the mapping (classifying) of all possible inputs to their correct output values. This convergence is achieved through supervised training with a large set of known input-output pair samples, or through unsupervised learning in which homogeneous clusters are self-organized from given input samples.

A typical neural network, shown in Figure 1, has three layers of processing units, termed nodes, representing input features, internal representations, and output values respectively. In this example, a node at any layer is connected to all nodes at the preceding layer and/or to all nodes at the succeeding layer. The state of the network at any instant is defined in terms of the activations of the nodes and the weights connecting these nodes. For the network shown in Figure 1:

$$x_j = f(\sum_{i=1} w_{ji} x_i)$$

where x_j denotes the activation (output) of a particular unit and w_{ji} is the weight of the connection between x_j and a node in the previous layer, x_i . The activation function, f, signifies the processing performed by a node on its input. This function can be an identity function, a threshold function (as in the case of Perceptron [24]), a logistic function (as in the case of backpropagation units), or some other function.

By adjusting the weights, it is feasible to get the desired output vectors for a given set of input vectors. Instead of manually adjusting the weights, backpropagation is a technique used to optimize the weights by minimizing the difference between the outputs and the corresponding target outputs over the entire set of input-target pairs. The process begins with an assignment of random values of weights, application of a set of input values, and computation of the output values. These values are compared with the desired output values, called target outputs, and the differences are computed, squared, and added together. Backpropagation minimizes the sum of these squares. Mathematically, if E denotes this sum, k denotes the output layer, then:

$$\Delta w_{kj} \propto -\frac{\partial E}{\partial w_{kj}}$$

Applying the chain rule:

$$\frac{\partial E}{\partial w_{kj}} = \frac{\partial E}{\partial in_k} \frac{\partial in_k}{\partial w_{kj}}$$

where in represents the inner product of the inputs with their corresponding weights. Noting that $\frac{\partial in_k}{\partial w_{kj}} = y_j$, applying the chain rule again, and solving, one gets for the output layer:

$$\delta_k = \frac{\partial E}{\partial in_k} = \frac{\partial E}{\partial y_k}\frac{\partial y_k}{\partial in_k}$$

$$\frac{\partial E}{\partial y_k} = -(y^* - y_k) \qquad \frac{\partial y_k}{\partial in_k} = f_k'(in_k)$$

For other layers:

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial in_j}\frac{\partial in_j}{\partial w_{ji}}$$

$$\frac{\partial in_j}{\partial w_{ji}} = y_i$$

$$\delta_j = \frac{\partial E}{\partial in_j} = \frac{\partial E}{\partial y_j}\frac{\partial y_j}{\partial in_j}$$

$$\frac{\partial y_j}{\partial in_j} = f_j'(in_j)$$

Therefore,

$$\sum_k \frac{\partial E}{\partial in_k}\frac{\partial in_k}{\partial y_j} = \sum_k \frac{\partial E}{\partial in_k} w_{kj} = \sum_k \delta_k w_{kj}$$

So,

$$\delta_j = f_j'(in_j)\sum_k \delta_k w_{kj}$$

where unit j is not an output unit.

The above technique provides recursive computation of weights by comparing the actual output with the computed output, and propagating error signals backwards through the network. Since corrections in the weights are proportional to the error in the output, the magnitude of the correction becomes smaller and smaller as the desired outputs are approached. This asymptotic approach to the final solution implies that the learning process becomes increasingly slower as the magnitude of the error reduces. As such, neural networks that use the backpropagation algorithm can involve a significant amount of training time, and a large number of training samples. Strategies for overcoming this problem are discussed later in this paper.

Backpropagation is one of the techniques used to train a neural network system to a particular problem environment. The learning process is dependent on the training samples. In a sense, neural networks learn the rules from these samples, unlike expert systems where human experts are critical for imparting this training.

In the next section, the process of automated reading is analyzed with the aim of identifying subprocesses that could benefit most from the application of evolving neural network technologies.

3 The Process of Automated Reading

Automated reading involves seven distinct processes. These processes, shown in Figure 2, are described below [6,13,31,34]:

i. Scanning: This is an optical process that provides a raster image of the document with sufficient spatial resolution and gray scale level to support further processing. The issue of gray scale level is more important for pictures and graphs, as compared to text. Also, while the optical scanner can distinguish between a large number of gray levels, thresholding mechanisms are employed to classify the scanned image into a smaller number of categories, in order to reduce processing bandwidth and memory requirements for other processes. However, this thresholding reduces accuracy, and creates complications in several situations.

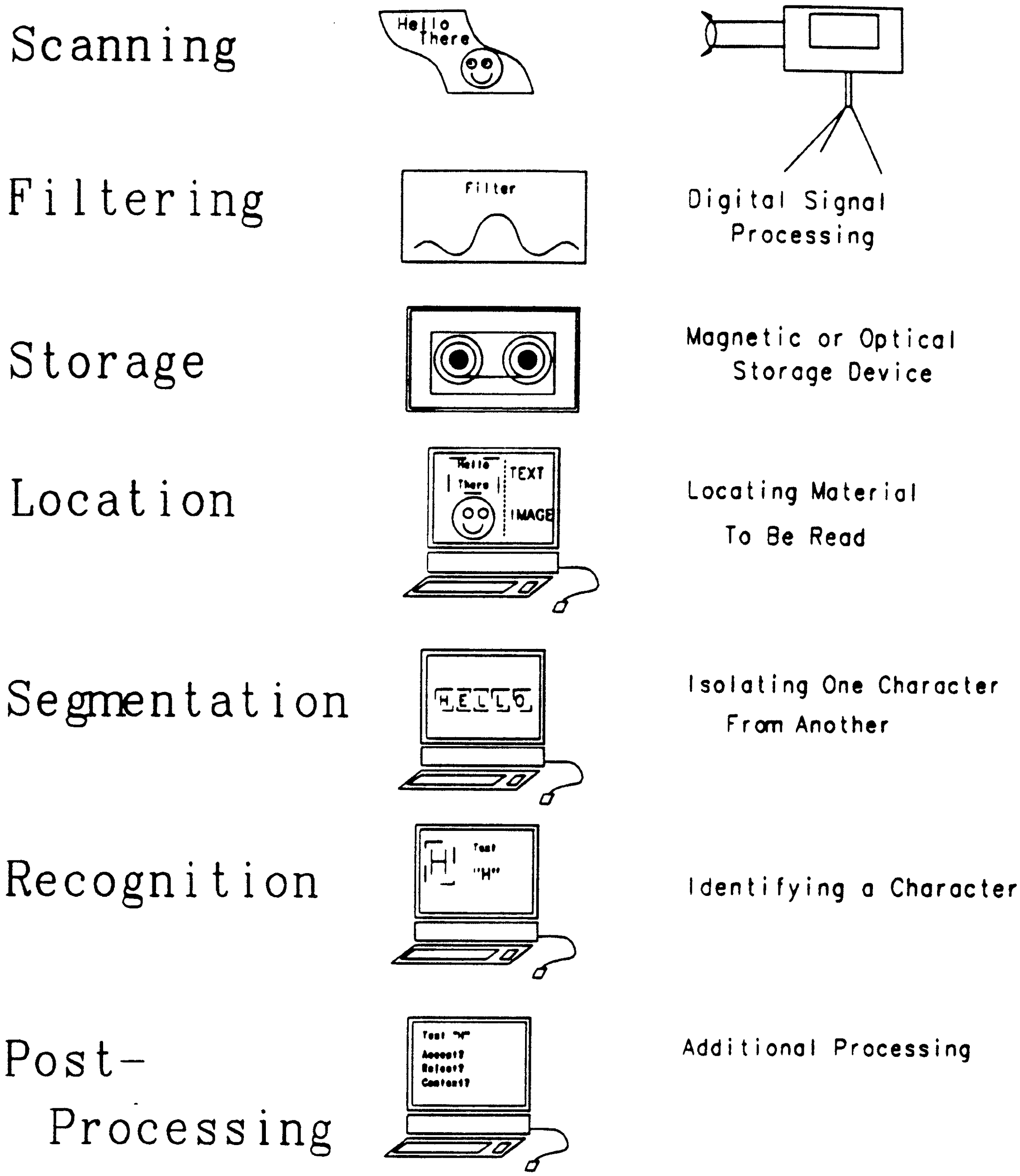


Figure 2: Key steps in automated reading

ii. Filtering: This process attempts to minimize the level of noise which originated in the source document or was introduced during the scanning process. Filtering enhances the quality of the image for easier recognition. In order to reduce the problem of varying thicknesses of lines in the handwritten material, "skeletonization" can be performed to make all line thicknesses uniform.

iii. Storage: The storage of information in a bitmap format requires a significant amount of memory. Even a single 8 1/2" x 11" sheet of paper scanned at 300 d.p.i., in single color and using a single threshold (black-white only), requires 1 MByte of memory space. Compression of data can reduce this requirement, but data in compressed form cannot be readily used for detecting and identifying patterns in the stored information.

iv. Location: This is the process of determining the location of the material that needs to be read. Further, information of different types (i.e., alphanumeric text, graphs, and pictures) must be distinguished and processed separately. Baseline drift correction techniques help to orient the courtesy amount relative to a horizontal scale in the case of checks.

v. Segmentation: This is the process of isolating one character from its immediate neighbors. In the case of continuous handwritten material, this is usually the most difficult process to accomplish. Even in the case of typed and printed material, adjacent characters spread into each other because of tight kerning, inadequate resolution of the scanner, poor quality of the document, or high brightness threshold. Defining a set of criteria for distinguishing between adjacent characters is difficult because of the many ways in which characters merge together and the fact that merged characters frequently contain misleading strokes.

vi. Recognition: Recognition of a character is based on defining and encoding a sequence of primitives that can represent a character as accurately as possible. In the template matching technique, the bitmap that constitutes the image of the character is compared with a succession of stored templates. Recognition of handwritten material generally uses the feature extraction technique in which one utilizes aspects such as pronounced angles, junctions and crossings, and properties of slope and inflection points. The choice of features is governed by their resilience to topological transformations, projective transformations, and translations.

vii. Postprocessing: In postprocessing, the goal is to decipher "rejects" and to correct "errors" using techniques such as contextual analysis and dictionary lookup methods.

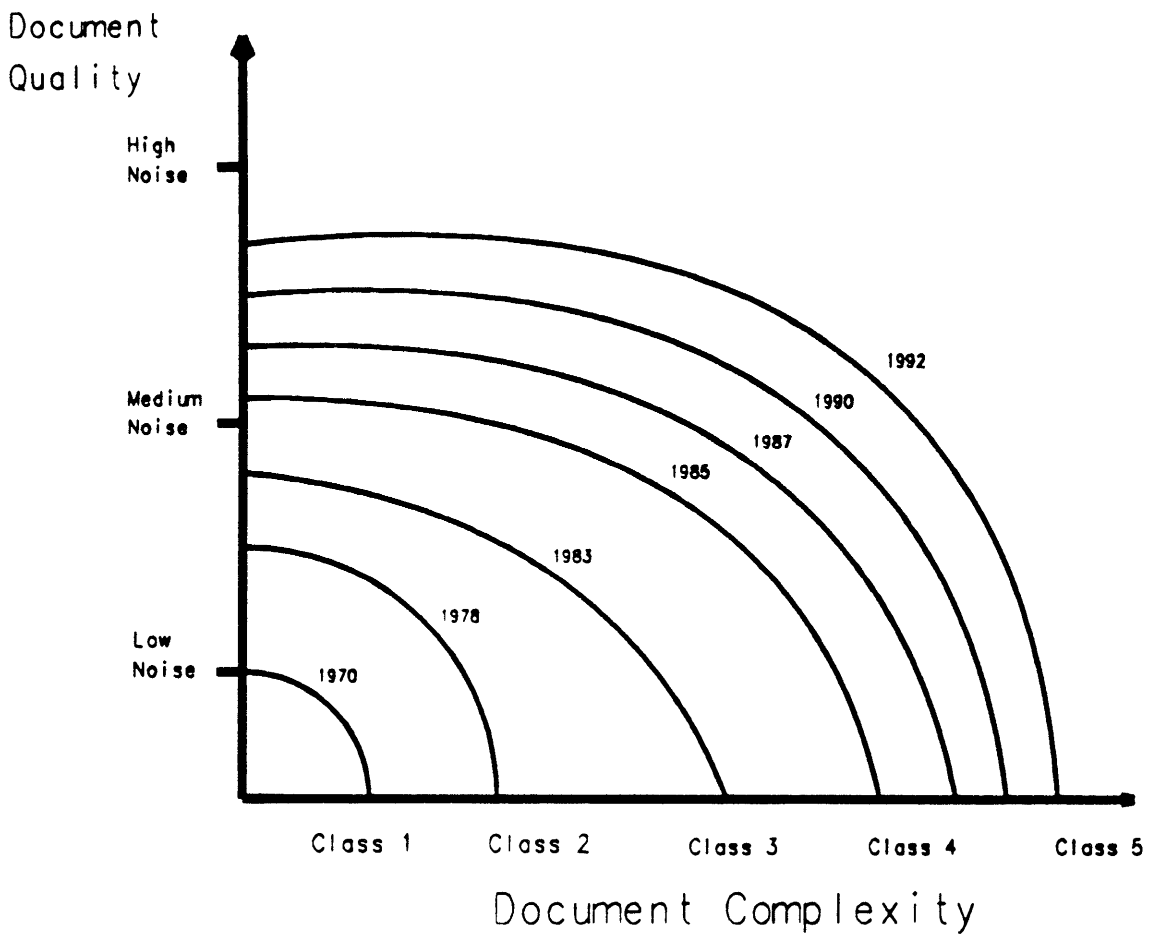
The above set of seven processes do not occur necessarily in the sequence stated here. In fact, multiple processes can occur in parallel with each other.

The different processes collectively determine the overall capability of the automated recognition. With respect to performance evaluation of alternative design approaches, there is no reliable way of modelling the accuracy of a reading machine except by comparison with a standard set of norms. The impracticality of statistical modelling is due to the fact that the pattern generating process and its multivariate statistics are influenced by a number of barely controllable, application-dependent parameters.

One taxonomy that attempts to predict the likelihood for automated reading of a given set of documents is described by Gupta et. al. in [11]. Documents are classified, based on their quality, into three classes: (i) Low noise documents; (ii) Medium noise documents; and (iii) High noise documents. After carefully examining a large number of documents from different fields, a five-class system was proposed for classifying documents based on their complexity, with Class 1 representing straight basic text-only documents, and Class 5 representing integrated documents with text and images, but no handwritten material. The evolution of technology and the capabilities to deal with an increasing range of documents is shown in Figure 3. Handwritten information is even more difficult to read than Class 5 documents, motivating the use of newer technologies to mitigate some of the difficulties.

4 Neural Network Approaches to Reading Handwritten Material

One of the landmark models that employs neural network technology for pattern recognition is Fukushima's Neocognitron model [8]. In this model, shown in Figure 4, pattern matching is achieved via nonsupervisory learning techniques (clustering methods). Weights between input and simple cells are modified such that the simple cell is trained to respond most strongly to patterns occurring most frequently in its receptive region. Translation, distortion, and slight rotation invariance are achieved via complex cells, each of which receives the outputs of a set of simple cells in the same layer. The firing of any simple cell in this set is sufficient to cause the complex cell



DOCUMENT COMPLEXITY

Class 1: Single column, monospaced, single pitch
Class 2: Single column, multifont, mixed spacing
Class 3: Single column, some images, formatted
Class 4: Multicolumn document, tables
Class 5: Multicolumn, mixed text and images

DOCUMENT QUALITY

Low Noise: Original typewritten or typeset document, clearly separated charcters, no skewing

Medium Noise: Easily readable photocopy or original laser print characters not touching

High Noise: Broken and touching characters, fading ink, skewed text

Figure 3: Evolution of scanning technology[11]

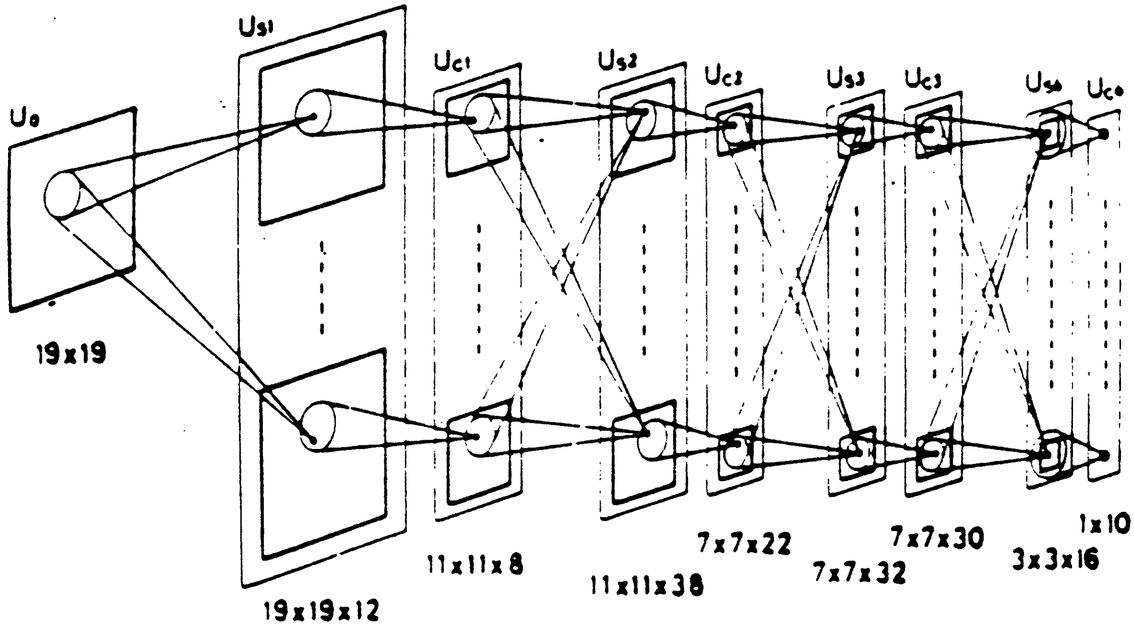


Figure 4: Fukushima's Neocognition model [8]

to fire. This implies that the complex cell is less position sensitive than any one of the simple cells. Multiple layers of simple and complex cells are present, with each layer having a more abstract response (i.e., each layer having fewer cells, and each responding to a larger neighborhood of cells) than the previous layer. At the output layer, each complex cell is a representation of a complete input pattern. This approach has been applied successfully to several applications, including character recognition. The one big disadvantage is that it requires a significant amount of computation to perform simulations on a conventional sequential machine.

In addition to recognition, recent versions of the Neocognitron model can segment digits[12]. Since feedback signals are sent back only from a unique output cell, only the signal components corresponding to one recognized pattern reach the recall layer, even when the input stimulus consists of two or more touching or overlapping digits. Therefore, the output of the recall layer can be interpreted as the result of segmentation, where only the components relevant to a single pattern are selected from the stimulus. Extensions of the Neocognitron model include both fine and coarse recognition networks [14]. The coarse network provides a focus of attention for the fine network. In one example[14], an 80 by 80 pixel input map is divided into 14 by 14 overlapping partitions, each receiving input from a 15 by 15 portion of the input map. Using a winner-takes-all inhibitory strategy in the top layer of the coarse network, one node can be activated (focused on) at a time.

A model reminiscent of the Neocognitron model has been utilized by designers at AT&T, in conjunction with traditional backpropagation (see Figure 5), to recognize postal zip codes [3,17]. Three hidden layers are used, with local averaging and subsampling between layers. These layers are designed to extract features of increasing complexity and to increase invariance with respect to distortions and translation of the input. The first hidden layer (H1) is composed of 12 groups of 64 units arranged as twelve independent 8 by 8 feature maps, each responsive to a different feature. Each unit of a feature map takes data from a 5 by 5 pixel area of the input image, with overlap of the receptive fields of the feature map. The second hidden layer contains 16 units arranged in a 4 by 4 plane. Each unit in H2 combines local information coming from 8 of the 12 feature maps in H1. Its receptive field is composed of eight 5 by 5 neighborhoods centered around units that are at identical positions within each of the eight maps.

As stated earlier, the backpropagation method of training tends to be slow. AT&T has speeded up the training process by using local connections and constraining weights corresponding to the same feature in alternative

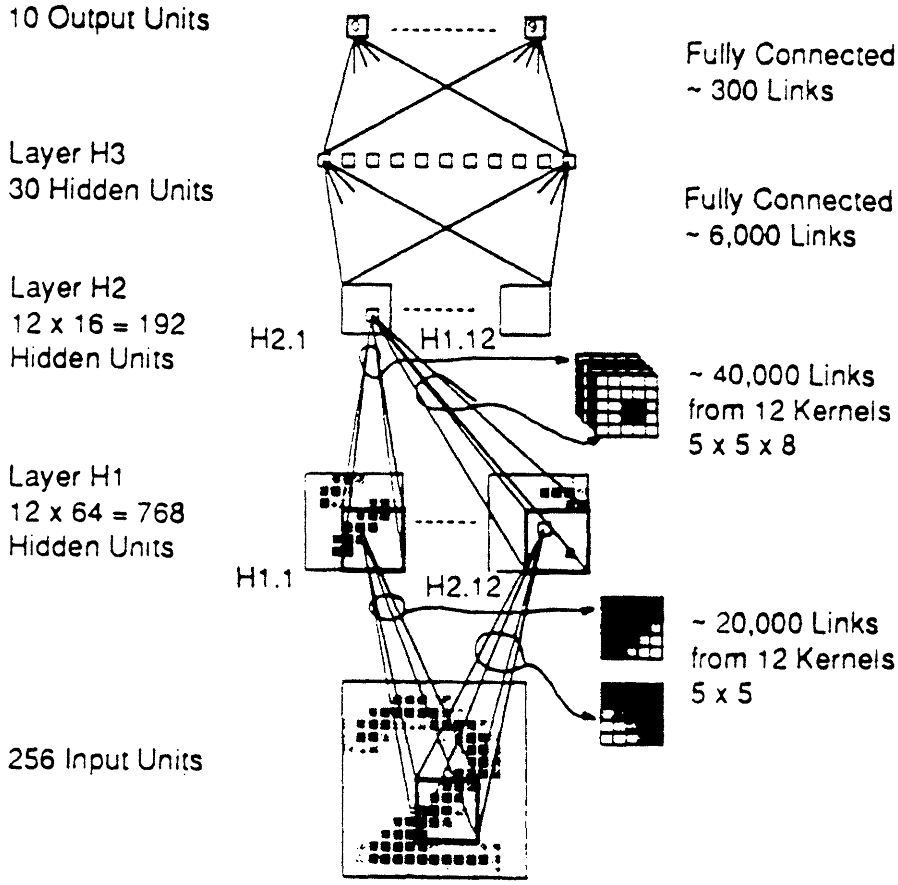


Figure 5: AT&T model [54]

areas of the image to be equal. This approach is similar to the traditional image-processing notion of a convolution filter smeared across the whole image to detect features such as edges. In the neural implementation, the filter coefficients are trained to extract (in the hidden layers) features of the size-normalized input pixel fields, which are useful in the recognition of the digits in the output layer. Because of constrained weights and local connections, there are fewer free parameters. Still, the learning of the weights using backpropagation required 30 passes through the training set, involving 3 days effort on a SUN Sparc Station equipped with special neural simulators. By training offline and then programming the parameters on digital signal processing chips, it is possible to achieve recognition speeds of 10-12 digits per second. A 1% error rate was achieved, with 9% of handwritten zip code samples being rejected [3].

The Neocognitron model also forms the basis for a remittance processing system developed by Nynex [8], and shown in Figure 6. In the Neocognitron model, the maximum output node can be found, but it may be with relatively low confidence. In the Nynex model, efferent connections are determined not only by the maximum output detector, but also by expected outputs and probabilistic update rules. Nynex uses various expected interpretations, each with its own initial probability based on apriori statistics, such as the amount of the bill, the amount of the previous bill, and the sum of these amounts. The probabilistic update rules are used to update the certainty values associated with the probability weightings of the expected interpretations. These updated values help determine the output cell used for feedback control, achieving faster convergence and handling more deformed inputs than if a maximum output detector alone was used.

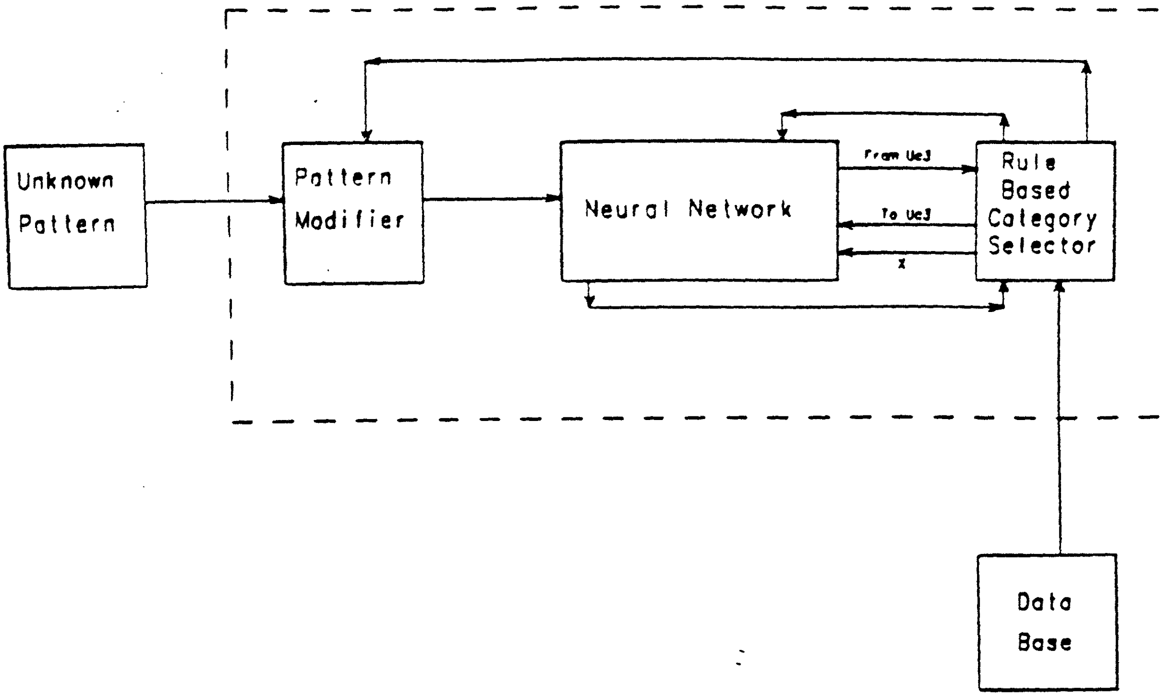


Figure 6: Nynex model [20]

This type of feedback control reinforcement is similar to adaptive resonance convergence in Grossberg's Adaptive Resonance (ART) model [10,36]. ART is able to self-stabilize its learning in a real-time unknown input environment while remaining plastic (changeable when a sufficiently novel input is presented). The stability and the plasticity are achieved, in part, through an exchange of efferent and afferent information between layers. This leads to a resonance in neural activity where features common to both the input and the expected model (digit) are reinforced. A gain control mechanism prevents efferent information alone (without any input) from causing neural activation. The two-thirds learning rule used in this system activates a neuron if at least two of the three (input, efferent, attentional gain) signals are present. Self-scaling is achieved, with small individual features being automatically given less weight than more complex input features. An orienting subsystem acts as a novelty detector, developing a new cluster for a sufficiently novel input vector, and controling the coarseness of categories. This is done by sending an inhibiting reset wave to a category which did not match the input well (i.e., if a "vigilance parameter" value is exceeded), enabling a new neuron to become active.

In the Nynex system, a rule-based expert system propagates certainty factors of the rules and their condition elements in a manner similar to the Emycin and M1 Expert System shells, and selects the output cell with the highest certainty for the purpose of feedback reinforcement. The processing is iterative, with convergence to a symbol indicating that its certainty has exceeded a certain threshold. The concept of a blackboard expert system architecture is utilized with independent knowledge sources being accessed depending on what is posted in a shared global data structure consisting of the candidate input, output, or intermediate level interpretations of the digits. By using a blackboard architecture, this system is more adaptive to variations in unconstrained handwriting than a task stack in which tasks are planned in a set sequence. The current version of the Nynex system can deal with fractional cents amounts in the courtesy number by locating the sign in expressions such as 32/xx, ignoring the denominator, and classifying the digits in the numerator. Two hundred features have been used. Palm trees and other background features of checks can be easily ignored with the Nynex system [47].

Unlike AT&T and Nynex who have based their designs on models reminiscent of the Neocognitron model, Nestor, Inc. has patented a restricted coulomb energy training method which is claimed by Nestor to eliminate the long training time and large training set requirements inherent in gra-

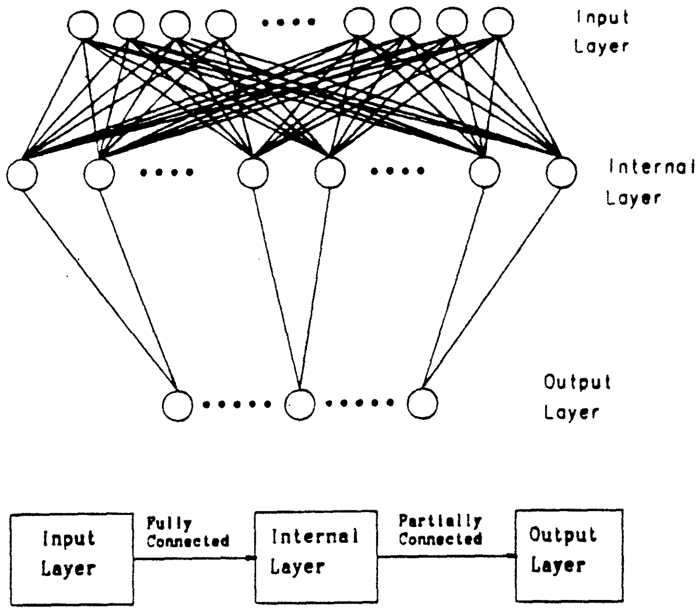


Figure 7: Nestor's RCE network model

dient descent paradigms [27]. Their Neural Learning System (NLS) scales well, and is able to define mappings supporting an arbitrary degree of non-linear separability. Because of the low connectivity and the use of integer arithmetic, off-the-shelf hardware such as transputers can be used [52]. In the patented version of the Nestor system, which has recently been applied to reading applications where segmentation is not an issue, three layers are used as shown in Figure 7. The weights connecting the first layer and the middle layer are fixed, and may be thought of as the centers of circular (or hyperspherical) subclusters, representing the prototype feature vector for the subcluster [27,30]. On receipt of error signals caused by a new input pattern, weights are not adjusted; instead either the thresholds (radii) defining the subclusters are modified or new hidden nodes (i.e. subclusters) are dynamically formed (cell commitment). If the system, in supervised learning, is given an input of a particular class which does not fall within a subcluster belonging to that class, a positive error signal commits a new cell and all inputs within a certain radius of the given input to that particular class. The threshold (radius) is lowered in response to a negative error signal indicating intersection with another class.

A detailed comparison of the three approaches considered above (AT&T, Nynex, and Nestor), as well as two other approaches (HNC and Neurogen), is presented in Table 1. This table contains all the data that the companies were willing to release for publication at this stage.

5 ADAPTIVE TECHNOLOGIES

The seven processes involved in automated reading possess some underlying commonalities. In the scanning stage, the image is captured using image scanners or video cameras, and colors and intensities are selectively filtered and encoded; in the noise reduction stage, area-dependent thresholding is applied to filter noise; in the storage stage, the image is compressed; in the location stage, only one portion is selected for further processing; and this process continues. Noise reduction, compression, segmentation, interpolation of data, template matching, and recognition are all filtering operations. In the case of handwritten material, since the relevant statistics of the receptive field filter windows are unknown and the variance is large, it becomes desirable to use adaptive filtering techniques.

	AT&T	Nynex	Nestor	HNC	Neurogen
Input					
pixel field dimensions	variable	variable	variable	variable	variable
Constraint problems handled					
segmentation	yes	yes	no	yes	yes
skeltonization	not needed	yes	no	yes	no
image shift	partial	yes	partial	yes	yes
rotation	partial	partial	partial	yes	yes
skewing	yes	yes	partial	yes	yes
scaling	yes	yes	partial	yes	yes
zero/one bit map	yes	yes	yes	yes	yes
gray scale bit map	yes	yes	no	yes	yes
cents					
no	n/a	no	no	yes	no
00	n/a	yes	yes	yes	no
xx	n/a	yes	no	yes	no
decimal	n/a	yes	yes	yes	yes
/ notation	n/a	yes	no	yes	no
Network Architecture					
Number of units	variable	n/a	variable	n/a	variable
Number of connections	variable	n/a	variable	n/a	n/a
Modularity	yes	yes	yes	n/a	n/a
preprocessing network	yes	no	yes	n/a	n/a
feature mapping	yes	no	yes	n/a	n/a
convolution network	yes	no	yes	n/a	n/a
averaging net	yes	no	no	n/a	n/a
postprocessing	no	yes	no	yes	yes
decision system	no	yes	no	n/a	yes
expert system	no	yes	no	n/a	yes
Backpropagation model	yes	n/a	no	n/a	n/a
fully connected	no	n/a	-	n/a	n/a
number of layers	6 (basic)	n/a	-	n/a	n/a
number of input units	variable	n/a	-	n/a	n/a
number of output units	10 (digit)	n/a	-	n/a	n/a
number of hidden layers	n/a	n/a	-	n/a	n/a
transfer functions	yes	n/a	-	n/a	n/a
adaptive weights	yes	n/a	-	n/a	yes
fixed weights	yes	n/a	-	yes	no
feed forward network	mostly	n/a	-	n/a	yes
recurrent network	some	n/a	-	n/a	no
Fukushima model	no	no	no	no	no
hierarchical	-	-	-	-	-
averaging/subsampling	-	-	-	-	-
unsupervised learning	-	-	-	-	-
RBCS	-	-	-	-	-
maximum detector	-	-	-	-	-
afferent connections	-	-	-	-	-
efferent connections	-	-	-	-	-
RBCS controlled	-	-	-	-	-
RCE model	no	no	19 networks	no	no
unsupervised	-	-	yes	-	-
supervised	-	-	yes	-	-
recurrent	-	-	yes	-	-
ART	no	no	no	no	no

	AT&T	Nynex	Nestor	HNC	Neurogen
Pattern feature detection	yes	yes	yes	n/a	n/a
horizontals	yes	yes	yes	n/a	n/a
verticals	yes	yes	yes	n/a	n/a
end points	yes	yes	yes	n/a	n/a
corners	yes	yes	yes	n/a	n/a
t-sections	yes	no	yes	n/a	n/a
+-sections	yes	no	yes	n/a	n/a
diagonals	yes	yes	yes	n/a	n/a
direction of contrasts	yes	no	yes	n/a	n/a
arcs	yes	no	yes	n/a	n/a
features of features	yes	yes	yes	n/a	n/a
Original architecture	yes	yes	no	n/a	n/a
Fukushima/backpropagation	no	n/a	no	n/a	n/a
Feature mapping networks	yes	n/a	yes	n/a	n/a
Fukushima/without max. detection with RBCS & expert system	no	n/a	no	n/a	n/a
Expert system incorporation	no	yes	no	yes	yes
for contextual information	no	yes	no	yes	yes
using certainty factor calculus	no	yes	no	n/a	n/a
Learning					
training time	3 days	12 hrs	2-12 hrs	n/a	2-12 hours
fixed	yes	yes	no	n/a	n/a
continuous(adaptive)	no	no	no	n/a	n/a
retrainable	yes	yes	yes	yes	n/a
stochastics	no	no	no	yes	n/a
annealing	no	no	no	yes	n/a
Output					
winner take all	secondary	no	no	yes	n/a
probabalistic	primary	yes	yes	yes	yes
rejections	yes	yes	yes	yes	yes
discarded	n/a	no	no	yes	yes
available	yes	yes	yes	yes	yes
rate adjustable	yes	yes	yes	yes	yes
Performance					
percent rejected	7%	n/a	8%	n/a	n/a
error rate	1%	n/a	<1%	n/a	n/a
number of classifications/sec	10-30	n/a	n/a	50	4 -120
Hardware					
External boards	AT&T DSP-32C	n/a	Transputer	Anza-Plus coprocessor	DSP
Sun	SPARC w/SNC simulator	yes	n/a	available	n/s
Digital	n/a	n/a	n/a	if desired	n/a
IBM	PC	yes	n/a	if desired	n/a
Software					
source language	C++, lisp	C	C	C	C
source code modifiable	yes	no	yes	yes	yes
system expandable	yes	yes	yes	yes	yes

[n/a = information not available; - implies not applicable]

Further, traditional statistical and syntactic pattern recognition methods, based on application of Bayes' Rule, cannot merge supporting or contradictory information from different sources or effectively deal with missing information. They assume that, for a given class 'w', feature 'x' and specific input distributions, the apriori probability P(w) and the class conditional densities P(x|w) are known. These are used to find the aposteriori probability P(w|x). The class conditional densities are hard to determine, unless one assumes the form of the density function and then estimates the parameters from known samples. Maximum likelihood methods assume the parameters are fixed but unknown, whereas Bayesian methods assume that they are random variables with apriori known distributions; Bayesian learning through sample observations sharpens the density function, causing it to peak near the true values of the parameters [4,5].

Since none of the standard ways of parametrizing unknown distributions are suitable for handwritten information, nonparametric techniques such as Parzen windows [9] and k-nearest-neighbors estimate techniques [4] become more relevant. Nestor's neural network is more efficient than these techniques[52], with classification performed in accordance with the class membership of nearest neighbors, the nearest prototype, or the nearest cluster center. Increasing the number of features may improve performance up to a certain point, but it also increases the cost and the complexity of the feature extractor and classifier.

Other nonparametric techniques assume a certain form of the discriminant function to separate the classes. In the Perceptron [24], a line is assumed to separate the classes, and hidden internal neural network layers are successively used to transform the original representation and to determine more complex separating boundaries. This process is continued until a final representation is attained in which the desired separation can be achieved with a hyperplane. The coefficients of the hyperplane are adaptable weights, playing an analogous role to the Bayesian conditional probability estimates. Neurogen, Inc. claims to possess a relatively linear digit recognition neural network with no hidden layer. This network is claimed to learn faster than backpropagation techniques, to offer stability, and to permit initialization with zero-valued weights rather than the usual random weights.[41]

Overall, the performance of statistical techniques must be compared with neural network techniques on a case-by-case basis. The Digit Recognition Applications Group of AT&T initially used Parzen windows and k-nearest neighbors, both of which are suited to parallel computer implementation and require no learning time. Unlike other statistical classifiers, the Parzen window method allows for continuous adaptation to additional inputs and to a changing environment. The designers at AT&T found that only with a high performance preprocessor and a large training database did the neural network surpass these techniques. Once trained, the neural network gave much faster classification and required less memory than the Parzen window technique[3].

Further, with neural networks, the distributed representation of features and other characteristics provides increased fault tolerance and classification even when noise is present. Distributed representations usually occur within localized modules [23], and efficiently utilize parallel hardware to implement best-fit searches, such as in associative memory implementations. Unlike traditional computer databases, the stored patterns do not exist anywhere. Instead, weights are stored, and represent plausible microinferences which are applied to the input to reveal patterns that satisfy the largest number of these constraints and contradict the least [23]. Distributed representations can enable simultaneous application of many partially fitting rules (weight constraints), each rule applied to the degree it is relevant[23]. This, in turn, can enhance the overall capability to perform correctly in diverse situations.

6 TRADITIONAL TECHNIQUES AND NEW TECHNOLOGIES

It is pertinent to emphasize here that backpropagation and other gradient descent methods are not radically new methods; instead they improve upon conventional techniques since the number of operations in backpropagation is proportional to the number of parameters rather than a square of the number of parameters. As such, neural networks enable optimization techniques to be applied to syntactic and statistical problems which were previously deemed to be numerically intractable.

Usually, statistical techniques are used to extract pattern primitives, defined in terms of local properties, and syntactic techniques use these primitives to recognize the whole pattern. Syntactic techniques by themselves have difficulty dealing with noise. Also, syntactic techniques are computationally more expensive than statistical techniques. However, unlike statistical techniques, they provide structural description (called "explanation facilities" in the terminology of expert systems), in addition to classification. In the syntactic approach, local templates are related to each other using context-free grammar rules which may be parsed using push-down automata methods. The templates are matched with the input digit, using inexact, error-correcting matching procedures such as relaxation. Relaxation and other constraint satisfaction techniques are beginning to utilize artificial intelligence concepts to incrementally update the knowledge base, minimizing inconsistencies and tolerating noise. The planning techniques of expert systems become relevant for complex vision systems. Statistical and syntactic methods may be combined together using stochastic grammars. These grammars are used when one pattern has two or more different structural descriptions making normal syntax analysis methods inapplicable. The ambiguity can be handled using statistical information on pattern distortion and noise and this is another area where expert systems and neural networks can play an important role.

Further, when classes overlap, as when the number 'seven' is written so that it looks like the number 'one', fuzzy set theory techniques can be used, as an alternative to having an additional neural network, to perform disambiguation. In [29], a fuzzy similarity relation between input digits is defined based on the distance of the digits and of template models from eight fixed points. Fuzzy set techniques provide a link between the statistical and logic methodologies, implemented using contemporary expert system tools.

Historically, the thrust was on bottom-up processing, that is, combine inputs to form intermediate representations, and combine these representations to recognize the digit. However, in recent years, there has been greater interest in top-down control. Top-down neural network processing has been proposed by Rumelhart and McClelland [33] who use feedback from the target levels to selectively enhance activation of units at a lower level, and by Grossberg in his Adaptive Resonance (ART) models. The Nynex remittance processing system uses top-down processing and expert system rule-based certainty factor propagation for utilizing knowledge of the expected courtesy amount value possibilities, with corresponding application logic and meta-knowledge for determining what amounts should be rejected by the system and given to a human operator. For a check presented without a coupon, the operator can specify the confidence thresholds, which determine the reject rate. On the other hand, Nestor does not utilize external information to establish confidence thresholds. Instead, it uses a different schema which combines the outputs of 19 neural networks for digit recognition, in order to select the correct output digit with greater accuracy.

One architecture for combining real-time parallel image processing with symbolic and neural computation is presented by Roman in [39]. Here, expert system rules call up image-processing and neural network routines and pass parameters from frame slots to the routines. Traditional, symbolic, and neural constructs for blackboards, abstraction, invariance, constraint optimization, confidence estimation, and explanation capabilities are discussed in [39]. Although applied in the context of target recognition, these ideas are applicable to handwriting recognition as well. Filtering, storage, location, and segmentation can be speeded up using multi-stage pipelined image processors such as the PIPE.

7 Optimization of Architecture

In our opinion, traditional image processing techniques, expert systems, and neural networks must be used in unison to enable courtesy amounts on bank checks to be read at high speed and with high accuracy. Multiresolution pyramid architectures, used in recent image processing approaches can be expanded to support cellular operations. Apriori knowledge (such as the expected amount and the various notations for the courtesy amount) should be embedded in symbolic rules, frames, and uncertainty paradigms, resident on interacting knowledge sources for the pyramid blackboard. This knowledge can provide least commitment control strategies to guide the application of inter- and intra-level image processing and relaxation operations.

While it may appear that neural network technology is a solution to all pattern matching problems, this is not true. In the infant stage of this technology, there are severe restrictions on the capabilities of neural networks, which can be overcome by selectively integrating it with established traditional signal processing, pattern matching techniques, and expert systems techniques. Neural network pattern classifiers come in many varieties, some of which were mentioned earlier in this paper. While the architecture and learning mechanisms of these models varies significantly, they all provide a training mechanism for "learning" a classification problem environment from examples of the data provided. Once trained, these networks are expected to classify any input within a given margin of error.

Our research into various neural network approaches reveals four facts. First, the performance of all the components in the seven processes described above plays an important role in the overall capability of the system. Although, no one component makes the system, one poor component can break it. Second, systems utilizing neural network technology generally perform better than systems using traditional image processing techniques alone, especially when the problem environment contains data which do not fit a predefined set of possible inputs, such as in the case of handwritten numbers. Third, no one neural network model appears to be inherently better than others to a significant extent; instead, technical superiority is attained by carefully tailoring various types of network models based on the needs of a particular problem environment. Fourth, the use of distinct classification schemes in parallel and the use of a voting scheme for image recognition increase the accuracy, the robustness, and the flexibility of the pattern recognition system.

Based on the above, a powerful classification system can be implemented with neural network models operating in parallel with other pattern classification techniques. Such a system offers the benefits of acceptable physical size and high accuracy. Running multiple classification systems in parallel permits for simultaneous design and production of several modules independently, facilitates expandability, and most importantly increases accuracy of classifications. The confidence of being correct increases when more than one classifier confers on a given classification. As such, a pattern recognition system can be optimized by utilizing multiple classifiers — neural network and non-neural network classifiers — working in parallel on the same input. Such a system, shown in Figure 8, involves two non-neural network recognition methods – dynamic programming and hidden Markov processes, and three neural network architechures – feature detection network, backpropagation, and recurrent backpropagation. The detailed design is in progress, and it is likely that additional classification modules would be utilized. In our design, the tentative result is fed into a post-processing expert system that incorporates context sensitive information. Also the idea of using an expert system to influence the behavior of a classifying network directly is being investigated.

In parallel with the above research activity, an enhanced structural tech-

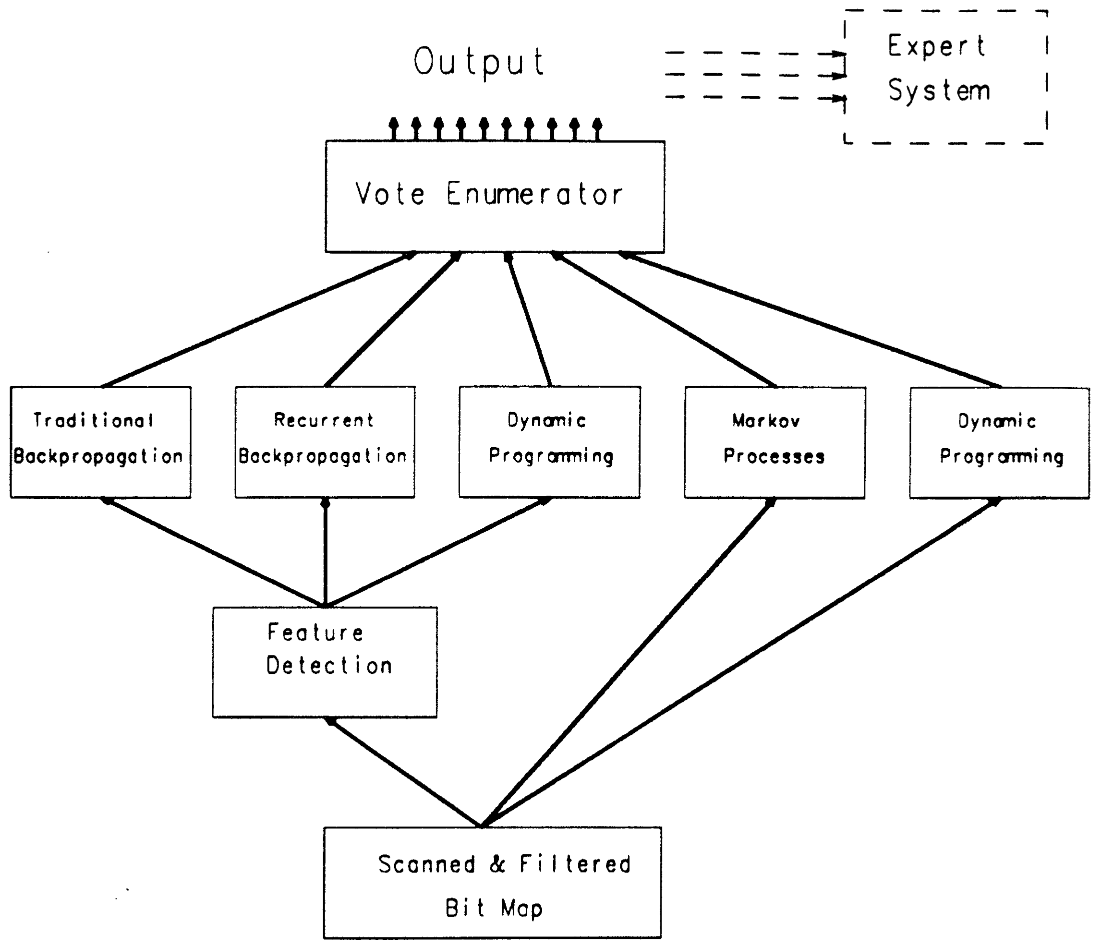


Figure 8: Recognition model based on multiple technologies

nique has been published in Wang and Gupta [37]. Current effort is focused on extending this technique to utilize neural network technologies to automatically optimize the size of the window used in this structural approach.

8 CONCLUSION

In the progression from recognition of printed material to recognition of unconstrained handwritten information, global template-matching and rule-based solutions become more complex, so that additional local information is needed to provide acceptable levels of accuracy. Auxiliary data play a more important role, and deterministic algorithms get replaced by heterarchical blackboard mechanisms and self-organizing neural network learning systems which range from simple arithmetic operations to automatic programming paradigms that can learn complex logic operations.

By combining appropriate image-processing, statistical, expert systems, and neural network approaches, it is feasible to enhance the capability for reading handwritten materials. Higher accuracy is made possible by improving segmentation and noise reduction capabilities, eliminating redundancies, identifying optimal feature combinations, defining operations invariant to translations and deformations, and adding knowledge to the system. Specifically, segmentation can be handled using image-processing and neural network techniques; shift invariance by global transformations and gradually increasing the size of the receptive fields in neural layers; noise problems by using smoothing filter operations, statistical and relaxation techniques, and distributed neural network representations; redundancy by skeletonization algorithms, statistical and neural network principal components analysis methods, and automated knowledge acquisition algorithms; and feature extraction by formulating expert system rules for representing knowledge of the interclass variations or by the neural network itself. Using a combination of several technologies, it is now becoming feasible to automatically read handwritten material, and research is continuing to achieve the high accuracy needed for critical applications.

9 ACKNOWLEDGEMENTS

One of the authors (E. Roman) visited AT&T, Nestor, Neurogen, and Nynex, as well as the OEM manufacturer, Computer Entry Systems, and had detailed technical discussions with their staff; the cooperation extended by these companies is sincerely appreciated. After the technical comparison matrix was initially created, it was sent to the first four companies and to Hecht-Nielsen Neurocomputers for verification of contents and for filling of blank cells; the time and effort spent by technical personnel in these five companies has vastly enhanced the quality of the comparison matrix. Instead of implementing a new system from scratch, the authors intend to design and develop the system in close cooperation with one or more of these five companies. The authors are indebted to the individuals listed at [40-52] for their valuable inputs.

REFERENCES AND BIBLIOGRAPHY

1. Baptista, G. and Kulkarni, K., "A High Accuracy Algorithm for Recognition of Handwritten Numerals", *Pattern Recognition*, Vol. 21, No. 4, 1988, pp. 287-291.

2. *Darpa Neural Network Study: October 1987 - February 1988*, AFCEA International Press, 1988.

3. Denker, J., et al., "Neural Network Recognizer for Handwritten Zip Code Digits", AT&T Bell Laboratories, pp. 323-333.

4. Devijver, P. and Kittler, J. (editors), *Pattern Recognition Theory and Applications*, Spinger-Verlag, Berlin, 1986, pp. 1-81.

5. Duda, R., et al., *Pattern Classification and Scene Analysis*, John Wiley & Sons, Inc. 1973.

6. Fisher, J., *Neural Networks Recognition and Electronic Imaging: A Revolution in Automating Pattern Recognition*, Neurogen, Inc., Brookline, MA, 1990. To be published.

7. Freeman, J., et al., *Neural Networks for Battle Management Applications*, Ford Aerospace Corporation, Technical Report #FAC-TN 1072, 1988.

8. Fukushima, K., et al., "Neocognitron: A Neural Network Model for a Mechanism of Visual Pattern Recognition," *IEEE Transactions on Systems, Man and Cybernetics 13(5)*, September/October, 1983, pp. 826-834.

9. Gelsema, E.S. and Kanal, L.N., *Pattern Recognition and Artificial Intelligence*, Elsevier Science Publishers B.V., North Holland, 1988.

10. Grossberg, S., *The Adaptive Brain: Volumes I & II*, Elsevier, Amsterdam, 1986.

11. Gupta, A., Hazarika, S., Kallel, M., Srivastava, P., "Optical Image Scanners and Character Recognition Devices: A Survey and New Taxonomy", Technical Paper #107-89, International Financial Services Research Center, Massachusetts Institute of Technology, 1989. (MIT Sloan School of Management Working Paper #3081-89).

12. Fukushima, K., "A Neural Network Model For Selective Attention in Visual Pattern Recognition", *Biological Cybernetics 55*, 1986, pp. 5-15.

13. Huang, J. and Chuang, K., "Heuristic Approach to Handwritten Numeral Recognition", *Pattern Recognition*, Vol. 19, No. 1, 1986, pp. 1v5-19.

14. Jakubowicz, O., "Neural Networks for Intelligence Analysis", *Signal*, April 1989, pp. 47-54.

15. Khanna, T., *Foundations of Neural Networks*, Addison-Wesley Company, Inc., 1990.

16. Lam, L. and Suen, C., "Structural Classification and Relaxation Matching of Totally Unconstrained Handwritten Zip-Code Numbers", *Pattern Recognition*, Vol. 21, No. 1, 1988, pp. 19-31.

17. Le Cun, Y., et al., "Handwritten Digit Recognition with a Back-Propagation Network", *Neural Information Processing Systems*, Vol. 2, 1990.

18. Le Cun, Y, "Handwritten Digit Recognition: Applications of Neural Network Chips and Automatic Learning", *IEEE Communications Magazine*, November 1989.

19. Leung, C., Cheung, Y., and Wong, Y., "A Knowledge-Based Stroke-Matching Method for Chinese Character Recognition", *IEEE Transactions on Systems, Man and Cybernetics*, Vol. SMC-17, No. 6, 1987.

20. Loris, K., and Euchner, J., U.S. Patent #4,876,731, October 24, 1989, pp. 1-12.

21. Mantas, J., "An Overview of Character Recognition Methodologies", *Pattern Recognition*, Vol.19, No. 6, 1986, pp. 425-430.

22. Martin, G. and Pittman, J., "Recognizing Hand-Printed Letters and Digits Using Backpropagation Learning", *MCC Technical Report, Human Interface Laboratory*, Austin, TX, January 1990.

23. McClelland, J. and Rumelhart, D., *Explorations in Parallel Distributed Processing: A Handbook of Models, Programs, and Exercises*, MIT Press, 1988, pp. 77-86.

24. Minsky, M. and Papert, S., *Perceptrons: An Introduction to Computational Geometry*, MIT Press, Cambridge, MA, 1969.

25. Pao, Y., *Adaptive Pattern Recognition and Neural Networks*, Addison-Wesley Publishing Company, Inc., 1989.

26. Rajavelu, A., et al., "A Neural Network Approach to Character Recognition", *Neural Networks*, Vol. 2, 1989, pp. 387-393.

27. Reilly, D., et al., "Learning System Architectures Composed of Multiple Learning Modules", *Proceedings of First Annual IEEE International Conference on Neural Networks*, 1987, pp. II495-II503.

28. Rifkin, G. "The Giants Focus on the Image of a Check," *The New York Times*, March 11, 1990, p. F9.

29. Devijver, P. and Kittler, J (editors), *Pattern Recognition Theory and Applications*, Spinger-Verlag, Berlin, 1986, pp. 383-391.

30. Scofield, C., et al., "Pattern Class Degeneracy in an Unrestricted Storage Density Memory", *Neural Information Processing Systems*, American Institute of Physics, 1988, pp. 674-682.

31. Shridhar, M. and Badrelding, A., "Recognition of Isolated and Simply Connected Handwritten Numerals", *Pattern Recognition*, Vol. 19, No. 1, 1986, pp. 1-12.

32. Soucek, B. and Soucek, M., *Neural and Massively Parallel Computers: TheSixth Generation*, John Wiley & Sons, New York.

33. Ref[23] ibid, pp. 209-219.

34. Gupta, A. and Kallel, M., "Automated Transfer of Information from Paper Documents to Computer Accessible Media", Working Paper #2113-89-MS, Sloan School of Management, M.I.T., April 1989.

35. Uhr, L., (editor), *Parallel Computer Vision*, Academic Press, Inc., Boston, 1987.

36. Vemuri, V., *Artificial Neural Networks: Theoretical Concepts*, Computer Society Press of the IEEE, Washington D.C., 1988.

37. Wang, P. S-P., and Gupta, A., "An Improved Structural Technique for Automated Recognition of Handwritten Information," International Financial Services Research Center Working Paper #123-90, Sloan School of Management, M.I.T., 1990. (Under review by International Journal of Pattern Recognition and Artificial Intelligence).

38. Ref. [37] ibid, p.5.

39. Roman, E., "Intergration of Parallel Image Processing with Symbolic and Neural Computations for Imagery Exploitation," *Proceedings of the SPIE Conference*, San Diego, August 1989.

40. Personal communication with J. Farmer, Hecht-Nielsen Neurocomputers, 5501 Oberlin Drive, San Diego, CA 92121.

41. Personal communication with A. Gingrande, Neurogen, 325 Harvard St., Suite 211, Brookline, MA 02146.

42. Personal communication with W. Gladin, AEG, 1350 Connecticut Ave N.W., Washington D.C., 20036.

43. Personal communication with J. Guthrie, Computer Entry Systems, 2120 Industrial Parkway, Silver Spring, MD 20904.

44. Personal communication with L. Hastings, Bank Administration Institute, 60 Gould Center, Rolling Meadows, IL 60008-4097.

45. Personal communication with D. Lee and C. Anderson, Science Applications International Corporation (SAIC), San Diego, CA.

46. Personal communication with Y. Lecun, J. Denker, and L. Jackal, AT&T Bell Labs, Holmdel, NJ.

47. Personal communication with K. Loris, J. Euchner, and J. Thomas, Nynex Corporation, White Plains, NY.

48. Personal communication with G. Martin, Microelectronic & Computer Technology Corporation (MCC), Austin, TX.

49. Personal communication with S. McCready, IDC Avante, Framingham, MA.

50. Personal communication with C. Wilson, National Institute of Standards & Technology, Gaithorsberg, MD 20899.

51. Personal communication with C. Wolfe, AGS, Inc.

52. Personal communication with D. Wright, C. Scofield, and L. Kenton, Nestor, Inc., One Richmond Square, Providence, RI 02906.