Neural Abstraction Pyramid: A hierarchical image understanding architecture

Sven Behnke and Raúl Rojas

Abstract— **A hierarchical neural architecture for image interpretation is proposed that is based on image pyramids and cellular neural networks and that is inspired by the principles of information processing found in the visual cortex. Algorithms for this architecture are defined in terms of local interactions of processing elements and utilize horizontal as well as vertical feedback loops. The goal is to transform a given image into a sequence of representations with increasing level of abstraction and decreasing level of detail. A first application, the binarization of handwriting, has been implemented and shown to improve the acceptance rate of an automatic ZIP-code recognition system without decreasing its reliability.**

Keywords— **image interpretation, cellular neural networks, hierarchy, image pyramids, horizontal and vertical feedback, binarization.**

I. Motivation

The goal oriented interpretation of visual stimuli is one of the main tasks of the human brain. Humans extract most of the information about the environment by active vision. The image interpretation performance of the visual system is impressive and exceeds by far the performance of todays technical systems. Humans are, for example, able to recognize faces in a fraction of a second, even if the recognition task is complicated by a change of perspective, lighting, facial expression, hair style or the aging of a person. These abilities are even more remarkable when considering the fact that the brain is composed of slow "processors", the neurons, that are unreliable and inaccurate.

However, these deficits are more than offset by the visual system's massively parallel architecture. Retina, corpus geniculatum laterale, and visual cortex are composed of neural layers that are connected locally. The layers form a hierarchy of several abstraction levels where feedback links are about as strong as feed-forward links.

In addition, the brain manages to focus the limited resources to the relevant visual stimuli. This is done by a mechanism, called attention control, driven by salient visual stimuli and the interpretation goal. The potential contribution of the extracted information to the solution of the task faced by the individual determines which visual stimuli are relevant in each situation.

If we were able to build technical systems that are oriented on the biological information processing principles present in the visual cortex, it should be possible to reproduce main features of the human visual perception, such as robustness, speed, and the disambiguation of stimuli by the use of contextual information and heuristic assumptions. This would open a wide range of possibilities for the application of information processing systems.

The remainder of the paper is organized as follows: In the next section we review the traditional approach of image interpretation as well as some efforts to overcome its limitations. The proposed Neural Abstraction Pyramid architecture is presented and some algorithmic principles that are based on this architecture are outlined in section III. Section IV exemplifies the use of the architecture on a simple application, the binarization of handwriting. Experimental results show that the proposed architecture can improve the performance of a ZIP-code recognition system. The paper concludes with a discussion of the results and gives an outlook to further work.

II. Image Interpretation Systems

Traditional image interpretation systems have been used for decades. Usually they go through the following steps:

- *Preprocessing:* Image processing operations, such as filtering, edge detection, and histogram operations, are applied to improve the quality of the given image.
- *Segmentation:* The image is decomposed into homogeneous regions that should correspond to objects.
- *Feature extraction:* A number of characteristic measures, such as Fourier coefficients, moments, or structural features, are computed for every object.
- *Pattern recognition:* Based on the extracted feature vector the object is assigned to a class.

Traditional image interpretation methods yield useful results if the interpretation task can be defined clearly and the image acquisition is done in a stabile and controlled environment. Examples for the successful application of these systems are found e.g. in the areas of quality control or recognition of printed text.

If the interpretation task can be specified only in a fuzzy way, or the images are acquired in a natural environment, the deficits of traditional systems become obvious. These systems are mostly based on a feed-forward information flow. The missing feedback prevents the systems from using a larger context in order to resolve local ambiguities.

Several attempts have been made to overcome these limitations. One is the development of methods for the *control of the system's attention.* Such methods determine the most relevant parts of an image and apply feature extraction and pattern recognition only to these. Some of these methods generate saccades: the so called focus of attention jumps from one interesting point to the next [1], [2]. Others use image pyramids that contain the image in several resolutions [3], [4]. Interpretation starts here at the coarsest

S. Behnke and R. Rojas are with the Institute of Computer Science, Free University of Berlin, Takustr. 9, 14195 Berlin, Germany.
E-mail: {behnke|rojas}@inf.fu-berlin.de
We would like to acknowledge the support of Siemens AG.

0-7803-4859-1/98 $10.00©1998 IEEE

resolution. Hypotheses about the existence of certain image features are generated at some locations of the coarsest resolution and are verified at more detailed resolutions.

Another approach is the *cellular neural network (CNN)* [5] method. These networks are composed of identical processing units that are usually arranged in a two-dimensional grid and are connected locally. Templates determine, how the processing elements interact with their neighbors. Within this framework powerful local operators have been defined that can be used for example for image improvement or segmentation. Among the algorithms that are based on local horizontal interaction of processing elements are *region-growing* [6], *anisotropic diffusion* [7], and *relaxation labeling* [8], [9]. One of the most attractive features of the CNN paradigm is that the computations can be carried out at high speed by arrays of very simple analog processors.

Horizontal interaction between processing elements limits the feedback of information to a single abstraction level. Consequently, there exist some image interpretation approaches that use *vertical interactions* between processing elements that are located at different abstraction levels [10], [11], [12]. Here, the information flow in the forward direction serves the detection of features, while the feedback is used for the propagation of hypotheses from a more abstract layer to a less abstract representation. By using this vertical feedback loop it is for example possible to complete broken contours of objects present in the image. It is even possible to create virtual edges at locations where no contrast exists. These results correspond to findings about certain visual illusions that emerge e.g. by looking at Kanizsa figures [13]. However, most of the existing models are limited to the interaction of only two levels of abstraction.

III. Neural Abstraction Pyramid

The proposed hierarchical neural architecture for image interpretation is based on the ideas of image pyramids and cellular neural networks and is inspired by the principles of information processing found in the visual cortex. Algorithms for the proposed architecture are outlined that are defined in terms of local interactions of processing elements that utilize horizontal as well as vertical feedback loops. The goal is to transform a given image into a sequence of more and more abstract representations while the level of detail decreases. Models that describe the potential image content facilitate the adaptation of the system to a specific application. An attention control focuses the limited resources to the parts of the image that are relevant for the interpretation task.

A. Architecture

The main features of the proposed architecture are:

- *Pyramidal shape:* The neural processing elements, called *nodes*, are arranged horizontally in two dimensional *layers* that are arranged vertically to form a pyramid. The bottom layer has the size of the input image and the number of nodes decreases by a factor of four at the next level.

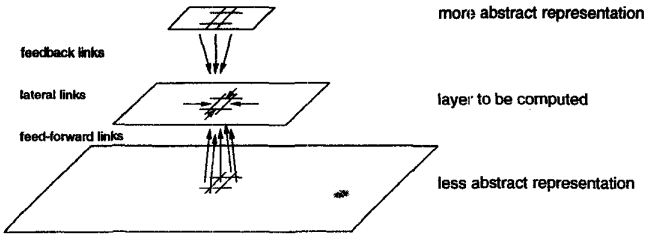

Fig. 1. Sketch of the proposed Neural Abstraction Pyramid.

- *Analog representation:* The nodes of each layer form a matrix and describe the image in a two dimensional representation. The level of abstraction of these representations increases with height, while the level of detail (spatial resolution) decreases. The bottom layer stores the given image (a signal). Subsymbolic representations are present in intermediate layers, while the highest layers contain descriptions of the image content that could be almost called symbolic. The representation consists of one or more *quantities*. These are stored in matrices of *state variables*, that can have a *value* from a finite interval. The reliability of such a value can be described by a *confidence* parameter while the importance of the knowledge of a certain state variable's value for the correct interpretation of an image can be expressed by a *need* parameter.
- *Local interaction:* Every node is connected to some nodes from its local pyramidal neighborhood via directed *weighted links*. The shared weights of all nodes in a layer are described by a common *template*. If the interpretation task requires the individual adaptation of weights, these are changed according to a common *adaptation template* that is restricted to use local information only. The following classes of links are essential to the interaction:
  - *Feed-forward links:* They lead from a lower layer to neighboring nodes of a higher layer and extract features. The tree that is rooted in a node defines its receptive field.
  - *Lateral links:* They connect neighboring nodes of the same layer and facilitate the consistent interpretation of the given image by means of processes, such as region-growing, relaxation labeling, and anisotropic diffusion.
  - *Feedback links:* They lead from a higher layer to a neighboring node in a lower layer and are used for the downward propagation of interpretation hypotheses.
- *Discrete time computation:* The update of a node's state variables for the time step (t) depends only on the input states at step ($t-1$). The new value can be computed as weighted sum of the input values that is passed through an output function, such as a saturated linear function or a sigmoidal. In addition, the confidences and need values are updated as well. Inputs with high confidence that are easy to interpret lead to results with high confidence while ambiguous inputs or inputs with a low confidence produce outputs with low confidence. High need values are propagated in the opposite direction from a node via the weighted input links to its input nodes. The computations are performed in one of the following sequences:

– *Layer by layer:* All nodes of a layer are updated simultaneously. The layers are processed in a predetermined sequence, e.g. from bottom to top.

– *Priority driven:* The update sequence depends on the presence of reliable inputs and the need of the results. Need and the possible confidence increase define a priority. The nodes with the highest priorities are updated first.

• *Multiscale representation:* Each quantity can be realized as image pyramid, e.g. via smoothed subsampling.

B. Algorithms

Due to the high flexibility of the proposed architecture a number of well-known image interpretation methods can be mapped efficiently to the data structure. Many local image processing operators, like linear filters, can be implemented by using only a single layer. Algorithms that have been developed for image pyramids can be mapped particularly well. They can work in a top-down as well as in a bottom-up mode. Furthermore it's easily possible to implement algorithms that are based on horizontal coupling of nodes.

The proposed architecture is not limited to the above methods, but it has been designed to facilitate the development of algorithms that utilize both horizontal and vertical feedback loops. Such computational processes are to be implemented in a way that honors the principles of *Gestalt psychology*, e.g. proximity, continuity, closure, and simplicity. To make this possible, it is necessary to specify for each layer simple consistent representations that model the objects potentially present in the image. The link weights and update rules have to be designed such that they favor simple and consistent representations against complicated or inconsistent ones.

The image interpretation algorithms work *iteratively*. As initialization the given image is fed into the bottom layer and assigned high confidence values. The top layer is initialized with high need values. In the course of computation the confidence values spread upwards while the need values are propagated downwards. By using vertical and horizontal feedback the image is interpreted first at locations where little ambiguities about the interpretation exist. These partial results are used in the following iterations to provide a larger context for the interpretation of the more ambiguous image parts.

The computation can be terminated if high confidence values are present at the top layer. Since at each step the quantities constitute a valid representation of the image, the termination may be done at any time, e.g. to meet real time requirements. However, the interpretation quality is expected to increase with the number of iterations.

In order to focus the system's limited resources to the most relevant image locations an attention control mechanism has been implemented on the basis of the confidences and need values. When using the priority driven update, only the paths in the pyramid that have high confidence and are needed for interpretation are calculated. The use of a multiscale representation allows to inspect the image parts at the resolution that is appropriate for the task.

Initially, the representations and templates have to be constructed manually. To simplify the modification of algorithms for a specific application, adaptation methods are needed. When developing learning algorithms for the adaptation of templates to a specific task, it is important to use only local information for the computation of weight updates. In contrast, global optimization methods, such as genetic algorithms and evolutionary techniques, can be used to optimize a given algorithm.

IV. An Application: Binarization of Handwriting

A simple application is used to illustrate the use of the proposed Neural Abstraction Pyramid architecture. The task is to separate handwriting in the foreground, defined by dark lines, from the paper, that constitutes the background. The gray level images used were provided by Siemens AG and show scanned ZIP-codes from large size letters (flats) that have to be sorted by the German postal service. Since the envelopes of these letters are mostly made of dark paper, the binarization task is nontrivial.

Histogram based thresholding techniques are among the most popular binarization methods described in the literature [14]. If the gray level histogram of the image is bimodal, the two peaks correspond to the background and the foreground, respectively. One can search for a local minimum in the smoothed histogram between the two peaks and use the corresponding gray value as a threshold for binarization. The main advantage of these methods is their low computational cost. On the other hand, histogram based methods fail, if a gray level gradient is present in the image, e.g. due to inhomogeneous illumination or reflections. Furthermore, the limitations of the histogram based methods become visible when lines are broken or strong noise is present in the image. Since they are not able to remove spot noise or to complete broken lines, the automatic ZIP-code recognition is complicated (see figures 9 and 10).

Here, we present a more powerful binarization method that is based on the Neural Abstraction Pyramid architecture. The idea is to detect and represent the lines and to assign their corresponding pixels to the foreground. The main assumption is: "Foreground is, where a line is present". The fact that lines in handwriting usually exhibit good continuity can be exploited.

Three levels of abstraction seem to be task relevant:

• Level 0: gray values, foreground/background separation
• Level 1: edges in four orientations
• Level 2: lines in four orientations

These levels of abstraction correspond to the layers of the pyramid. Note that edge elements cover 2×2 pixels, while line elements correspond to 4×4 pixels of the original image. In addition, some of the quantities are implemented as image pyramids. In these cases subsampled and smoothed versions are stored in the higher layers of the pyramid. The distribution of the representations in the pyramid is depicted in figures 2 and 5.

Currently, only a single value is used for the representation of a quantity. These values are interpreted as confidences about the presence of a certain feature at the corre-

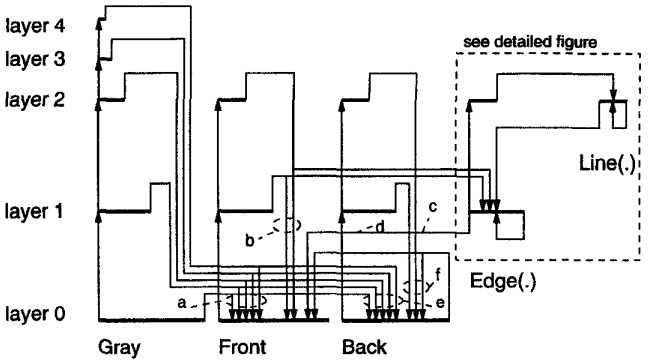

Fig. 2. Sketch of the binarization application. The distribution of the quantities in the pyramid is shown as well as the link pattern.

(a) Gray (x, y, .):	z	z+1	z+2	z+3	z+4
	8	-2	-2	-2	-2
(b) Front (x, y, .):	z+1	z+2			
	0.2	0.2			
(c) SumEdges (x, y, z+1):	2				
(d) Back (x, y, z):	-2				
bias:	-0.1				

Fig. 3. Template Front. All Links to input values and the template bias are shown. For each link the input quantity, the relative input coordinates and the link weight are specified.

sponding image position. The update of the values $v_{(x,y,z,q)}$ at position (x, y, z) for quantity q is done as follows:

$$v^{t+1}_{(x,y,z,q)} = \sigma\left[\sum_{j\in\mathcal{L}(i)} \mathcal{W}(j) v^{t}_{(\mathcal{X}(j,x),\mathcal{Y}(j,y),\mathcal{Z}(j,z),\mathcal{Q}(j))} + \mathcal{B}(i)\right].$$

$i = \mathcal{T}(q, z)$ is the template that is associated with the quantity q at layer z, $\mathcal{L}(i)$ is the set of links of that template, and $\mathcal{B}(i)$ is the template bias. $(\mathcal{X}(j,x), \mathcal{Y}(j,y), \mathcal{Z}(j,z), \mathcal{Q}(j))$ describe location and quantity of the input value for link j, and $\mathcal{W}(j)$ is the link weight. The output function $\sigma(.)$ is here a saturation function that limits the values to the interval $[0, 1)$. The values are stored with a precision of 8 bits and initialized to zero. The interaction of the quantities is described by templates:

- *Gray value:* The original gray values of the given image are stored in quantity Gray of layer 0. The layers 1 to 4 contain subsampled and smoothed versions.
- *Foreground and background:* The differences between the levels of the Gray pyramid indicate the presence of contrast of a certain scale in the image. We are interested in the high frequency portions of these contrasts that correspond to the image details. The template Front computes the image details that are darker than their surround (see figure 3) while Back contains the image details that are brighter than their surround (see figure 4). Note that the templates relate the Gray value of a pixel not directly to surrounding pixels, but to the upper nodes in the Gray pyramid (a, e). Thus, the immediate neighborhood of a node is inspected more closely than the more distant surrounding. Both templates have links that facilitate a region-growing process by excitatory feedback from upper levels (b, f) in order to produce active regions that have smooth borders. In addition,

(e) Gray (x, y, .):	z	z+1	z+2	z+3	z+4
	-8	2	2	2	2
(f) Back (x, y, .):	z	z+1	z+2		
	0.2	0.2	0.2		
bias:	0.1				

Fig. 4. Template Back.

Front receives inhibitory input from Back (c) and excitatory input from SumEdges (d). These links strengthen responses that are supported by detected edges and remove the noisy ones that are in the background.

- *Edges:* The edge representation is distributed across four quantities Edge(–), Edge(|), Edge(/), and Edge(\) that correspond to oriented edges. In addition, a quantity SumEdges represents the sum of all four orientations and a quantity MultiEdges signals the existence of more than one edge at the same location (see figure 7). Figure 6 shows as an example the template for the computation of the horizontal edges. The links perform the following functions:
  - *Edge detection:* The template looks at two levels of the Front values (g, h) in order to detect edges.
  - *Cooperation with edges of similar orientation:* Excitatory links from edge elements that would form a good continuation (i, j, k) support such configurations.
  - *Cooperation with lines of the same orientation:* An excitatory link from the corresponding line element (m) strengthens edges that are supported by a line of the same orientation.
  - *Competition with edges at the same position:* Inhibitive input from MultiEdges (n) and SumEdges (l) prevents multiple edge responses for the same position and constrains the edges to a width of about one node.
- *Lines:* The line representation is in the same way distributed across four quantities. Again, the sum and indicators for multiple lines are computed. The horizontal line template is shown in figure 8. The links perform the following functions:
  - *Line detection:* The template receives excitation from edges that have the same (q) or a similar (r, s) orientation.
  - *Cooperation with lines of similar orientation:* Lines that form a good continuation strengthen each other via excitatory links (t, u, v).
  - *Competition with lines at the same position:* Multiple line responses are inhibited by the corresponding MultiLines detectors (x). Finally, lines are constrained to a width of about one node via negative feedback from SumLines to positions that are orthogonal to the line orientation (w).

For each iteration the templates are evaluated from bottom to top. Note that in the first iteration the links from higher layers do not contribute to the result. Figures 9 and 10 show the computation after one, two, and five iterations for some ZIP-codes. It can be seen that most lines are detected initially, but some of the Front responses are caused by noise and some lines are broken. In the course of the computation the relevant Front responses, supported by detected edges and lines, are strengthened and completed while the noise responses are suppressed.

It is difficult to asses the quality of the binarization with-

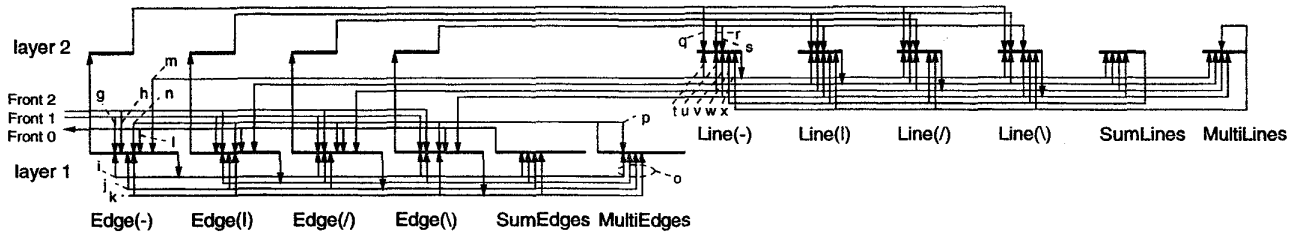

Fig. 5. Detail of the binarization application.

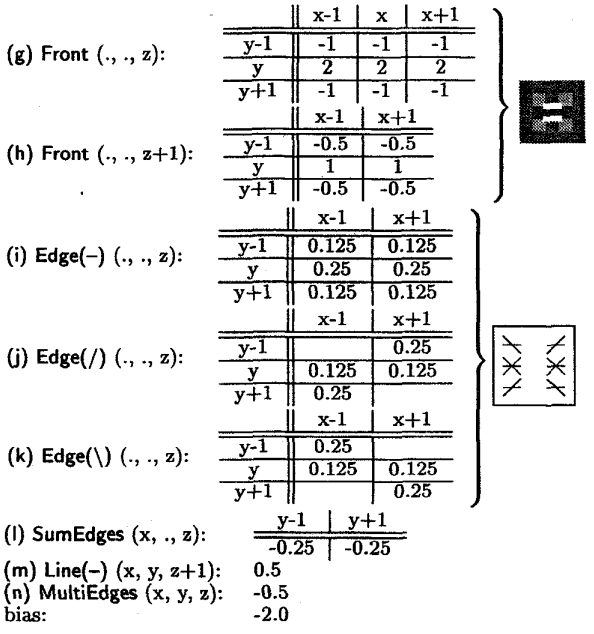

Fig. 6. Template Edge(–).

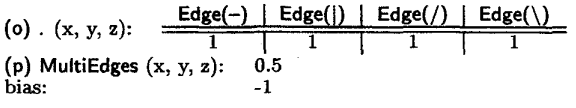

Fig. 7. Template MultiEdges.

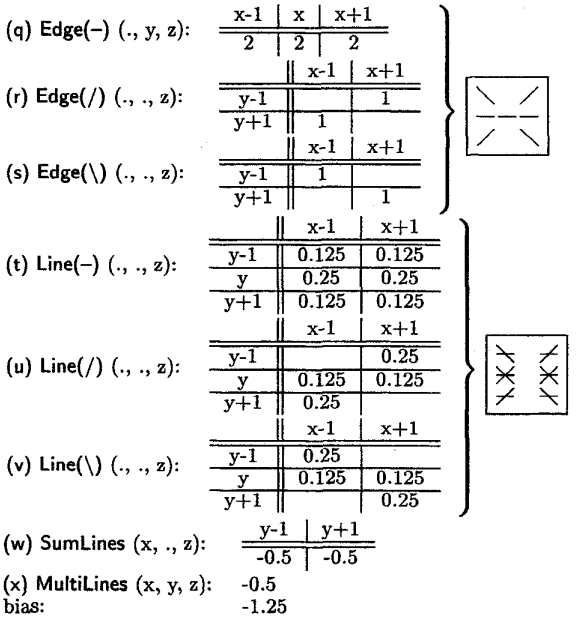

Fig. 8. Template Line(–).

out considering the system that uses the binarized images as input. Therefore, we investigated how the performance of a ZIP-code recognition system that is already used in a large scale real world application can be improved by a more powerful binarization.

A set of 503 images has been selected from a database of 4134 five digit ZIP-code images in the following way. The images were binarized using a histogram based thresholding technique. Then, a recognition system [15] was run to determine the ZIP-code. If that system rejected the image for the reason of low confidence, but the confidence was above a threshold, the image was selected. Thus, the selected images are difficult to recognize, but there is still hope to determine the correct ZIP-code. If the original system would have accepted these images, 289 of them would have been substitutions (differing from the true ZIP-code at least at one position). The selected images are then binarized using the proposed method and again presented to the recognition system. If the system accepts the image and the ZIP-code for the second run corresponds either to the best or to the second best ZIP-code from the first run, the image is accepted. Otherwise, the image is rejected. The original system had an acceptance rate of 84.56% with 1.17% of the accepted ZIP-codes being substitutions. With the described modification 169 images were additionally accepted, while only one of these was substituted. Thus, the overall acceptance rate improved to 88.65% without decreasing the reliability of the system.

V. Discussion

In this paper a hierarchical neural architecture for image interpretation has been proposed. One of the main features of the architecture is the transformation of the given image to a sequence of representations with increasing level of abstraction and decreasing level of detail. When going from the bottom of the pyramid to the top, the semantic content of the representations changes from "where" toward "what". Due to the reduced spatial resolutions in the higher layers, local interactions between processing elements correspond to the use of contextual information from a large area in the original image. The architecture is adequate for algorithms that utilize both horizontal and verti-

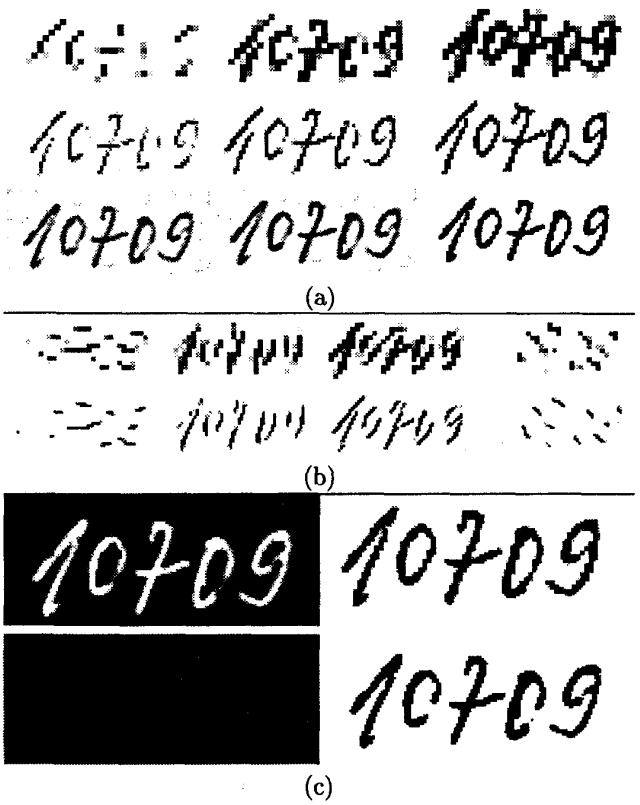

Fig. 9. Binarization of a handwritten ZIP-code: (a) quantities from left to right: after one, two, and five iterations; bottom to top: **Front**, **SumEdges**, and **SumLines**, (b) after five iterations from left to right: (–), (|), (/), and (\); bottom to top: **Edge**(.), **Line**(.), (c) after five iterations **Back**, **Front**, the original image, and the image where the background has been removed by thresholding.

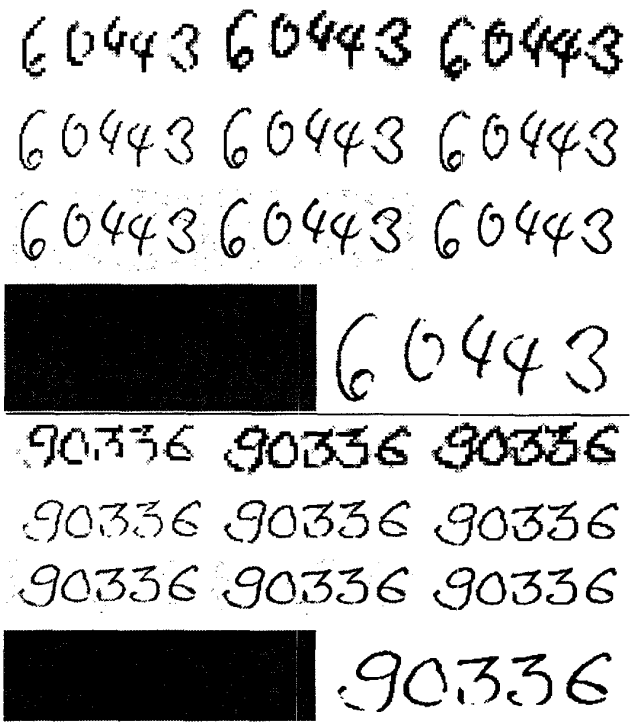

Fig. 10. Binarization of handwritten ZIP-codes. See figure 9(a, c).

cal feedback loops for cooperation and competition between representations.

A first application, the binarization of handwriting, has been implemented and has been shown to be perform well on images in which other methods fail.

So far, the templates have been designed manually, and no attention control has been implemented. We plan to implement learning algorithms and priority driven update for the binarization application in the near future.

This paper is part of on-going work in which the Neural Abstraction Pyramid will be used for more complex tasks, like the detection of regions of interest in letters and the recognition of handwriting. For these complex applications more than three layers of abstraction will be needed to represent the image content.

References

[1] Winfried A. Fellenz and Georg Hartmann, "Preattentive grouping and attentive selection for early visual computation," in *Proceedings of International Conference on Pattern Recognition*, 1996, pp. 340–345.

[2] Ernst Niebur and Christof Koch, "Control of selective visual attention: Modeling the "where" pathway," in *Advances in Neural Information Processing Systems*. 1996, vol. 8, pp. 802–808, The MIT Press.

[3] Steven Connelly and Azriel Rosenfeld, "A pyramid algorithm for fast curve extraction," *Computer Vision, Graphics, and Image Processing*, vol. 49, no. 3, pp. 332–345, Mar. 1990.

[4] Virginio Cantoni and Marco Ferretti, *Pyramidal Architectures for Computer Vision*, Advances in computer vision and machine intelligence. Plenum Press, New York, 1994.

[5] Leon O. Chua and Tamas Roska, "The CNN paradigm," *IEEE Transactions on Circuits and Systems*, vol. 40, no. 3, pp. 147–156, 1993.

[6] Rolf Adams and Leanne Bischof, "Seeded region growing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 6, pp. 641–647, 1994.

[7] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.

[8] Azriel Rosenfeld, R.A.Hummel, and S.W. Zucker, "Scene labeling by relaxation operations," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 6, pp. 420–433, 1976.

[9] Shih-Cheng Yen and Leif H. Finkel, "Identification of salient contours in cluttered images," in *Proceedings of Computer Vision and Pattern Recognition 1997*. 1997, pp. 273–279, IEEE.

[10] S. Grossberg and E. Mingolla, "Neural dynamics of perceptual grouping: Textures, boundaries, and emergent segmentation," *Perception and Psychophysics*, vol. 38, pp. 141–171, 1985.

[11] Heiko Neumann, Wolfgang Sepp, and Petra Mössner, "Adaptive resonance in V1-V2 interaction: Grouping, illusory contours, and RF-organization," in *Computational Neuroscience: Trends in Research, 1997, Proceedings Annual Computational Neuroscience Conference, Boston*, New York, 1997, pp. 753–757.

[12] W. von Seelen, S. Bohrer, J. Kopecz, and W.M. Theimer, "A neural architecture for visual information processing," *International Journal of Computer Vision*, vol. 16, pp. 229–260, 1995.

[13] G.K. Kanizsa, *Organization in Vision*, Praeger, New York, 1989.

[14] J Kittler and J Illingworth, "Minimum error thresholding," *Pattern Recognition*, vol. 19, no. 1, pp. 41–47, 1986.

[15] Sven Behnke, Marcus Pfister, and Raúl Rojas, "Recognition of handwritten digits using structural information," in *Proceedings of International Conference on Neural Networks, Houston*, 1997, vol. 3, pp. 1391–1396.