

Eye Identification for Face Recognition with Neural Networks

Åge Eide¹), Christer Jahren¹), Stig Jørgensen¹), Thomas Lindblad²),
Clark S. Lindsey²) and Kåre Østerud¹)

¹Ostfold College, Halden

²Royal Institute of Technology, Stockholm

ABSTRACT

The problem of facial recognition from grey-scale video images is approached using a two-stage neural network implemented in software. The first net finds the eyes of a person and the second neural network uses an image of the area around the eyes to identify the person. In a second approach the first network is implemented in hardware using the IBM ZISC036 RBF-chip to increase processing speed. Other implementations in hardware are also discussed, and includes preprocessing using wavelet (packet) transforms.

Keywords: wavelets, neural network, eye identification, hardware implementation

1. INTRODUCTION

The identification of individuals using face recognition represents a challenging task with many applications [1-3] in everyday life as well as in high security applications. Since the human face will vary in appearance in the short time as well as in the long time range, the inherent "slack in operation" of neural networks together with the redundancy and possibility to generalize are suggestive for implementing such a recognition system. If the system is implemented in dedicated hardware, rather than in software on a von Neumann computer, one will also benefit from the massively parallel neural network and obtain a real time system.

Recently, Rowley *et al* [4] presented a neural network-based system in which a small window of an image is examined to find and identify a face. The present work starts from a slightly different point, but is otherwise similar in the approach. Below we will present a system using a two-stage neural network. The first neural network starts with an image as input. The image is assumed to contain a face and the task of this net is to localize the eyes. A second neural network will use the information from the windows on the eyes then to recognize the person.

The total problem of identification is thus based on the assumptions: (i) that a person faces a TV camera, (ii) the eyes are localized by the first neural network and (iii) identified by the second network to belong to a known individual. We feel that this approach is typical for admission control, etc.

2. A "STRAIGHTFORWARD" NEURAL NETWORK APPROACH

As mentioned above, we need two neural networks: one to localize the eyes in the image obtained from the TV-camera and a second net to make the identification. In a first approach a 256 greyscale image of 360x280 pixel was obtained using a Panasonic NV-R30 video camera coupled to a frame grabber card (Screen Machine II) in a 100 MHz 486 personal computer. However, images were also produced using a HP flatbed scanner and photos.



Figure 1. Example of facial image with scanning rectangles of 13 x 9 pixels over the eyes.

2.1 Finding the eyes

The first stage neural network adopted a conventional feedforward network with one hidden layer. The network had 117 inputs and 15 hidden nodes and one output node. It was trained using the backpropagation algorithm using a sigmoid or tanh as transfer function. One reason for this choice is the potential implementation using the ETANN chip from Intel [5]. Two ETANNs can implement a 128-64-64 network, i.e. a system with 128 inputs and 64 outputs (and 64 hidden neurons). The chip is an analog, and will thus require DAC preprocessing. However, such implementations have been made previously [6] and found to work satisfactorily. Other commercially available chips are the ZISC radial basis function chip with 36 nodes [7], and the ALS430 from American Neuralogix [8]. All these chips are available with pertinent software to run on IBM personal computers and we have previous experience of implementing these systems [6, 9-13]

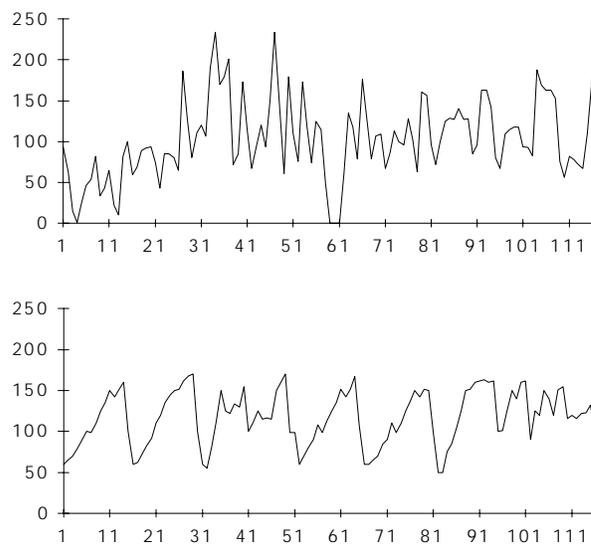


Figure 2. Input vectors from a window on a persons eye (top) and from another part of her face (bottom). The window has 13 x 9 pixels and the data is plotted linearly.

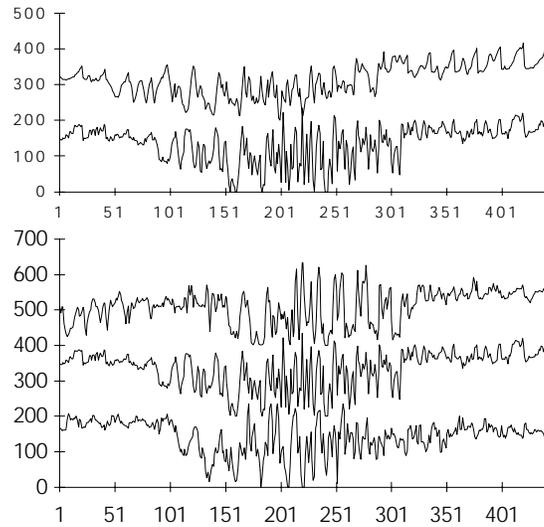


Figure 3. Two input vectors from the same person to the neural network trained for identification (top), and three input vectors of the eyes of three different persons (bottom). The vertical scale is in the 256 shades of gray (displaced 200 units for clarity).

The network used 373 images for training and 141 independent images for testing. The windows with 13x9 pixels (hence 117 inputs) are shown in fig. 1. Examples of input vectors are shown in fig. 2. Typically vectors from windows outside the face, are significantly smoother than vectors from the eye region. Provided that the window is not too big, these vectors turn out to have gross properties that are very similar. A relatively small window (13 x 9 pixel, cf. fig. 1) was used to scan for the eyes. The reason is that the eyes of different persons are most similar in the centre of the eyes, and this first stage needs only to locate the eyes. The network turned out to work very well. In a test of the redundancy of the network, we used an image taken at a factor of two shorter distance to the camera. Although the network was trained to find the eyes in an image recorded at a given distance, there was no difficulty in finding them in this extreme close-up. We also successfully used images recorded using scanners to test the neural network.

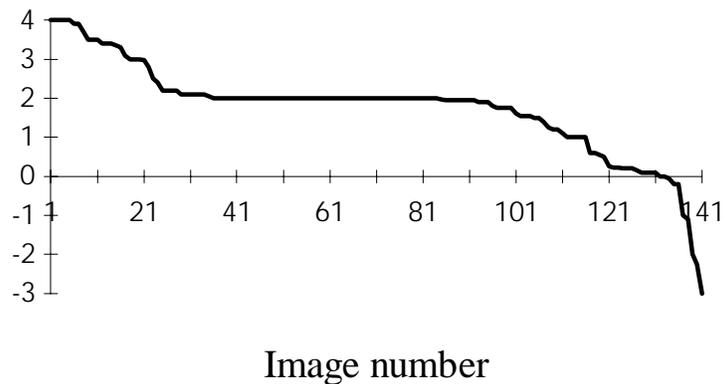


Figure 4. Results of testing the neural network using 141 images. The x-axis show the image number. The images are plotted in descending order with respect to the output from the neural net (y-axis). Negative values will thus represent misidentified cases.

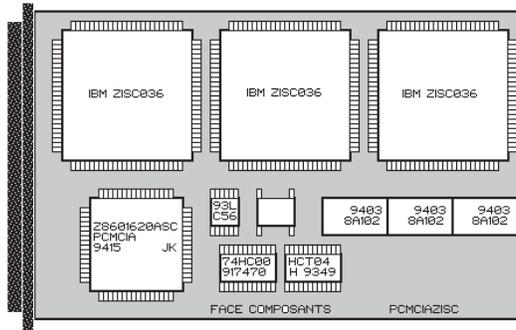


Figure 5. The implementation of three IBM ZISC036 RBF chips on a PCMCIA card (refs. [14,15])

2.2 Identifying the person

While the first stage neural network finds the common properties of eyes, the second one should find the differences. This neural network is of the same architecture but has 441 input nodes, 30 hidden nodes and has as many output nodes as individuals to be recognized. The larger number of inputs is due to the larger window used here. A square window of 21 x 21 pixels was used rather than the rectangular 13 x 9 pixel. Once again the available hardware had some influence on the choice of the size. The Intel ETANN multi chip board (MCB) can hold a maximum of eight chips and thus we can have 512 inputs. Similarly, the IBM ZISC is available on an ISA-card with 16 chips with 36 nodes each, i.e. again > 500.

The input to this network is shown in fig. 3. Here we show two input vectors from the same person (top) as well as the inputs from three different persons (bottom). It is quite clear that the eyes are most similar in the centre. This is the reason why we use a smaller window (13 x 9) for the network trained to find the eyes and a larger one (21 x 21) for identification.

The neural network was tested using 141 images of 25 persons. The result is shown in fig. 4, where we have plotted the output value of the node representing the desired output minus the second largest node output. The desired value is +1 for the node corresponding to the right person and -1 for all other nodes. This applies to both eyes. Hence the largest difference between any two outputs will be 4. In this way the vertical axis of fig. 4 will be a measure of how "close" the second largest output is (values between 0 and 4) or, in misclassified cases, it will show a negative value. As can be seen from the plot in fig. 4, approximately 5% of the images were wrongly identified.

The drawback with the suggested system is that when run on a Pentium personal computer it is too slow, and when implemented in hardware it is too large and/or expensive. Clearly, one would like to reduce the number of inputs by using some kind of feature extracting preprocessor.

A PCMCIA card, developed by Neuroptics Consulting [14] for GIAT [15] holds three ZISC036, suggesting that the total number of nodes should be kept below one hundred. In such a case it would be possible to run a real-time system on a lap-top computer. The CCD camera is easily interfaced to this bus. Below we will discuss some methods to reduce the number of inputs, while, at the same time, increase the flexibility and the redundancy of the total system.

3. A MORE "ADVANCED" APPROACH

Although the above system can be implemented using the Intel ETANN and/or IBM ZISC036 chips and thus perform in real time, there are still some inherent limitations, besides size and cost. One of them has been touched upon in connection with the task of finding the eyes, and has to do with the positioning of the face (distance, tilting, etc). For this

purpose one may choose another input to the aforementioned neural network. With the hardware in mind, one may choose to use the two projections of the window together with a histogram of the greyscale. In fact, the choice will depend on "how intelligent" the CCD camera is, i.e. how much preprocessing is done before sending the data to the neural network. In the suggested case we may have $9 + 13 + 10 = 32$ inputs, i.e. the 256 greyscales are binned in ten bins. Finding the eyes is very time consuming on von Neumann computers, and this is where you would benefit from a hardware implementation. Such implementations have been made by Neuroptics Consulting [14] on a PCMCIA card with three ZISC developed for GIAT Industries [15]. We have used this system to test the tracking of a person's eye in a short video sequence. The scanning window was 16×16 pixels and histogramming was performed using 64 bins. This test shows that simplicity as well as speed is gained in this manner.

Another approach to reducing the number of inputs is to use a wavelet (or wavelet packet) transform (cf. e.g. ref. [17] preprocessor. The WTP chip [16] from Aware Inc. is the only commercial hardware to our knowledge at this time (Sept. 1995). This processor computes both the forward and inverse wavelet transform (any 2, 4 or 6 coefficients system) on one-dimensional data streams. The circuit will then be used to calculate the transform, and the largest coefficients are retained and fed to the neural network doing the identification. However, some exploratory calculations [18] using both wavelet and wavelet packet transforms show that one do not get as sparse input data as one would expect. In fact, approximately 50% of the largest wavelet coefficients need to be retained in order to yield a fair reconstruction of the input vector (cf. fig. 7). Still this is quite useful in view of the available neural network chips.

4. SUMMARY AND CONCLUSIONS

The present paper has demonstrated that a two stage neural network can be used to find the eyes of a person and to identify him/her. Such a system could be run on a von Neumann computer, although the procedure would be irritatingly long. Significant reduction of the processing time can be obtained by implementing the neural network(s) in hardware. This may require systems with many inputs and in order to reduce this number an hardware preprocessing using wavelet (packet) transforms may be considered. The method of finding the region of interest (the eye) and identifying it, will be applicable to many areas (diagnostics, physics experiment, etc; ref. [19]) and very useful when implemented in hardware. The wavelet preprocessing turn out to be even more efficient when identifying impulses in telecommunication noise [20].

5. ACKNOWLEDGEMENTS

The present work was performed with support from the Swedish Engineering Research Council (TFR) under contract Dnr 240-93-400, which is gratefully acknowledged. We would also like to acknowledge discussions with IBM, Corbeil Essonnes, France, GIAT Industries, Toulouse and Neuroptics Consulting, Montpellier. In particular we acknowledge discussions on programming the ZISC with G. Paillet. The collaboration of the students of Ostfold College is acknowledged.

6. REFERENCES

1. R. Linggard, D.J. Myers, and C Nightingale, and Neural Net works for vision, speech and natural language, Chapman & Hall (1992)
- 2.I. Solheim, T. Payne and R. Castain, The potential in using backpropagation neural networks for facial verification systems, WINN-AIND, Auburn, AL, USA (1992)
- 3 I. Solheim, Neural Networks for Facial Recognition, Thesis, University of Tromsø (1991)
- 4 H.A. Rowley, S. Baluja and T. Kanade, Human Face Detection in Visual Scenes, CMU preprint July 1995: CMU-CS-95-158
5. Intel 80170 NX ETANN. Data Sheet, Intel Corp. Santa Clara, CA

6. B. Denby, Th. Lindblad, C. S. Lindsey, Geza Szekely, J. Molnar, Åge Eide, S.R. Amendolia and A. Spaziani, Investigation of a VLSI neural network chip as part of a secondary vertex trigger Nucl. Instr. Meth. **A335** (1993) 296 - 304
7. ZISC03 Databook, IBM, Essonnes, France
8. ALS420. Data Sheet, American NeuraLogix, Lake Mary, FA
9. J. Molnar, G. Székely, Th. Lindblad, C.S. Lindsey, B. Denby, S.R. Amendolia, Åge Eide, A Flexible VME-module with a 68070 Computer for Embedded Applications of the 80170NX Analog Neural Network Chip, ICFA Instr. Bull. No.10, Dec.1993.
10. C.S. Lindsey, Th. Lindblad, J.R. Vollaro, G. Székely, J. Molnar, Performance of a Cascadeable Neural Network VME-module with Intel 80170NX Chips, Nucl. Instr. & Meth., A351 (1994) 466-471.
11. Th. Lindblad, C.S. Lindsey, M. Minerskjöld, G. Sekniaidze, G. Székely, Åge Eide, Implementating the New Zero Instruction Set Computer (ZISC036) from IBM for a Higgs Search, Nucl. Inst. & Meth. A357 (1995) 217.
12. M. Minerskjold, Th. Lindblad, C.S. Lindsey. A versatile VME Digital Neural Network for Control and Pattern Recognition Applications. Proc. EANN'95: Int. Conf. on Engineering Applications of Neural Networks, Aug. 21-23, 1995 Otaniemi, Finland.
13. Th. Lindblad, C.S. Lindsey, Å . Eide, The IBM Zero Set Instruction Computer ZISC036, A Hardware Implemented Radial Basis Function Neural Network, to be in CRC Industrial Electronics Handbook, in press.
14. Guy Paillet, Neuroptics Consulting, Montpellier, France
15. G. Laurens, GIAT Industries, Toulouse, France
16. Wavelet Transform Processor Chip, User's Guide. P/N 2250, Aware, Inc. Cambridge, IL
17. I. Daubechies, Ten lectures on Wavelets, Soc. Ind. and Appl. Math. Press, vol. 61 CBMS-NSF, Philadelphia, 1992
18. F. Majid, R. R. Coifman, M. V. Wickerhauser, "The Xwpl system Reference Manual", Yale Univ.
19. Th. Lindblad *et al*, to be published
20. Th. Lindblad *et al*, to be published