A Pre-Processing Technique Based on the Wavelet Transform for Linear Autoassociators with Applications to Face Recognition

F.YANG[1], M.PAINDAVOINE[1], H.ABDI[2,3]

[1] LE2I-Université de Bourgogne, 6 Bd. Gabriel, 21004 Dijon France
email: fanyang@u-bourgogne.fr
[2] LEAD-CNRS-Université de Bourgogne
[3] The University of Texas at Dallas (U.S.A)

Abstract. In order to improve the performance of a linear autoassociator (which is a neural network model), we explore the use of several pre-processing techniques. The gist of our approach is to store, in addition to the original pattern, one or several pre-processed (i.e. filtered) versions of the patterns to be stored in a neural network. First, we compare the performance of several pre-processing techniques (a plain vanilla version of the autoassociator as a control, a Sobel operator, a Canny-Deriche operator, and a multiscale Canny-Deriche operator) on an example of a pattern completion task using a noise degraded version of a face stored in an autoassociator. We found that the multiscale Canny-Deriche operator gives the best performance of all models. Second, we compare the performance of the multiscale Canny-Deriche operator with the control condition on a pattern completion task of noise degraded versions (with several levels of noise) of learned faces and new faces of the same or another race than the learned faces. In all cases, the multiscale Canny-Deriche operator performs significantly better than the control.

Linear auto-associative memories are one of the most simple and well studied neural-network model[1] [2]. They are widely used as models for cognitive tasks as well as pattern recognition, or digital signal processing[3]. Even though linear autoassociators are known to be quite robust when noise is added to the patterns to be recognized, their performance is rather bad when a *lot* of noise is added to the stimulus. One of the ways of improving performance could be to use some pre- and post- processing of the patterns to be recognized. In this paper we explore the use of some pre-processing techniques using wavelet transform applied to multiscale edges detection. This paper is organized as follows. In the first part, the basic features of linear associative memories are briefly described. In the second part, we evaluate the performance of some pre-processing techniques. Finally, we present the results of some simulations evaluating the performance of the multiscale Canny-Deriche operator on a pattern completion task using faces as stimuli.

1   Description of linear associators

The class of models presented in this section are known as linear associators[3] [5]. They come in two forms: hetero- and auto-associators. The hetero-associator can be used to learn arbitrary associations between input and output patterns. The auto-associator is a special case of the hetero-associator in which the association between an input pattern and itself is learned. In this paper, we will consider only the linear auto-associator.

The advantage of linear associators in comparison with non-linear models is that they provide for the integration of a very large number of cells in the network. Their implementation is quite easy, because they can be analyzed in terms of the singular value decomposition of a matrix[3][5]. Besides, linear models constitute a first processing stage for numerous applications using more sophisticated approaches (see [3], [4] for reviews). In our description, we follow closely the formulation detailed in[5]. The patterns to be learned are represented by $L \times 1$ vectors $\mathbf{a}_k$ where k is the stimulus number. The components of $\mathbf{a}_k$ specify the values of the pattern to be applied to the L cells of the input layer for the k-th stimulus. The complete set of K stimuli is represented with a $L \times K$ matrix noted $\mathbf{A}$ (i.e., $\mathbf{a}_k$ is the k-th column of $\mathbf{A}$). The $L \times L$ synaptic weight connection matrix between the L input cells is denoted $\mathbf{W}$. Learning occurs by modifying the values of the connection weight between cells as explained later in this section. The response of the model to a pattern $\mathbf{x}$ (which may or may not have been learned) is obtained as $\widehat{\mathbf{x}} = \mathbf{W}\mathbf{x}$. Because auto-associators are generally interpreted as content addressable memories, their performance is evaluated by comparing the output of the system with a test pattern which can be a copy or a degraded version of one of the patterns previously learned by the system. This is achieved by computing similarity measures (most of a time a cosine) between input and output. The larger the similarity between input and output, the better the performance.

In order to achieve a high level of performance, several iterative learning rules have been proposed. The most popular one is clearly the Widrow-Hoff learning rule. This is an iterative procedure which corrects the connection matrix $\mathbf{W}$ using the difference between the target response and the actual response of the network. In matrix notation, the Widrow-Hoff rule is written as:

$$\mathbf{W}_{[t+1]} = \mathbf{W}_{[t]} + \eta(\mathbf{a}_k - \mathbf{W}_{[t]}\mathbf{a}_k)\mathbf{a}_k^T \tag{1}$$

with $\mathbf{W}_{[t]}$ being the weight matrix at step t, η being a small positive constant, and the index k being chosen randomly. The linear associator has been often analyzed in terms of the eigenvalue decomposition or singular value decomposition of a matrix. In terms of the singular value decomposition, the rectangular stimulus matrix $\mathbf{A}$ is decomposed as: $\mathbf{A} = \mathbf{P}\boldsymbol{\Delta}\mathbf{Q}^T$, with $\boldsymbol{\Delta}$, diagonal matrix of singular values ($\boldsymbol{\Delta} = \boldsymbol{\Lambda}^{1/2}$); $\boldsymbol{\Lambda}$, diagonal matrix of non zero eigenvalues of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$; $\mathbf{P}$, matrix of eigenvectors $\mathbf{A}\mathbf{A}^T$; and $\mathbf{Q}$, matrix of eigenvectors $\mathbf{A}^T\mathbf{A}$. The Widrow-Hoff learning rule can be analyzed in terms of the singular value decomposition of matrix $\mathbf{A}$ [3]. Specifically, $\mathbf{W}_{[t]}$ is expressed as[5]: $\mathbf{W}_{[t]} = \mathbf{P}\{\mathbf{I}-$

$(\mathbf{I} - \eta\boldsymbol{\Lambda})^t\}\mathbf{P}^T$. When η is smaller than $2\lambda_{\max}^{-1}$ ($\lambda_{\max}$ being the largest eigenvalue of $\boldsymbol{\Lambda}$), the Widrow-Hoff learning rule will converge to: $\mathbf{W}_{[\infty]} = \mathbf{P}\mathbf{P}^T$ which is the value we used in this paper. The matrix $\mathbf{P}$ is a $L \times N$ matrix with N being the number of non zero eigenvalues. Typically, L is significantly smaller than N (*i.e.*, $N \ll L$). As a consequence, using $\mathbf{P}$ directly instead of $\mathbf{W}$ will lead to an important gain in processing speed as well as storage. For example, when dealing with a face recognition application, the matrix $\mathbf{W}$ was a 33975×33975 matrix whereas the eigenvector matrix $\mathbf{P}$ was only a 33975×200 matrix.

2 A pre-processing using multiscale edges

The goal of learning is to find values for the connections between cells such that the response of the model approximates the input as well as possible. To assess the performance of the model, degraded versions of the previously learned stimuli are presented to the model as a test. If learning has been successful, then the response pattern will be more similar to the original pattern than the degraded stimulus was (see [1] for an illustration). In other words, autoassociators can act as pattern completion devices. In order to improve the performance of the auto-associator, we decided to explore the possibilities of storing one or several filtered versions of the patterns to be stored in addition to the original patterns. We refer to this technique as *pre-processing*. As we are mainly interested with image patterns, we choose filtering techniques meaningful in the this context. Because it is generally agreed that edges are essential for recognition[6], we decided to increase their importance in the image. Quite a number of algorithms have been proposed in the literature to perform edge extraction. We decided to implement three algorithms:

1. the *Sobel* operator (a differential operator) as it is considered as a standard procedure well suited for noiseless images.
2. the *Canny-Deriche* operator because it is known to be optimal for edge extraction in noisy images [7][8].
3. a multiscale resolution wavelet version of the Canny-Deriche operator, called the *multiscale Canny-Deriche* filter, also referred to as the *wavelet transform*[8]. It has been suggested that this technique should be more efficient than a one-scale resolution [9]. It is a separable filter when applied to 2D images. Its impulse response for a 1D signal (because of its separability, the filter can be seen as two 1D filters) is given by:

$$f(x) = ksxe^{msx} + e^{msx} - e^{sx} \tag{2}$$

with $k = 0.564$, $m = 0.215$, and where $s = 2^j$ is the scale factor (with, in our case, $j \in \{0, 1, 2, 3\}$), and x being the pixel position. This method is implemented as a wavelet transform using a convolution between the image and the edge detection filter for different scales ($s = 2^j$). As a result, this filter detects edges occuring at different scale resolutions in the image [9].

In order to compare these different techniques, we implemented four models:

1. a standard auto-associator storing 40 different face images 225×151 [3],
2. an auto-associator storing the original 40 face images plus, for each face, a Sobel filtered image of the face (hence a total of 80 face images),
3. an auto-associator storing the original 40 face images plus, for each face, a Canny-Deriche filtered image of the face (hence a total of 80 face images),
4. an auto-associator storing the original 40 face images plus, for each face, four multiscale Canny-Deriche filtered face images (one face image per scale resolution, hence a total of 200 images).

All the models were tested using the same procedure. A random face (previously learned) was chosen, and random noise was added to it before presentation.
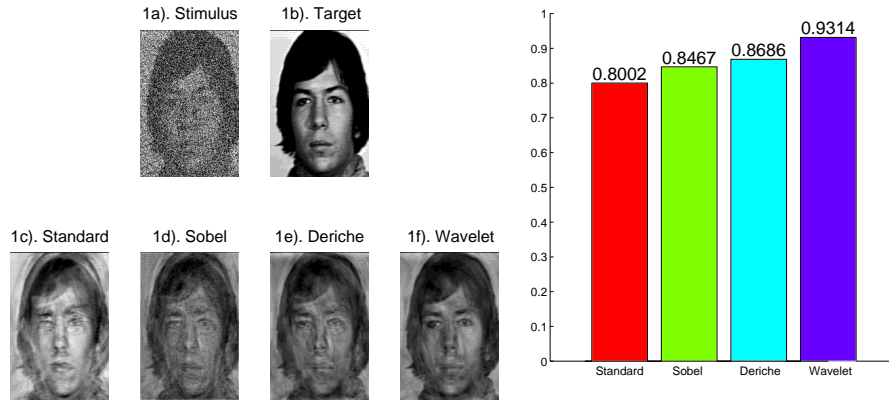

Fig. 1. Response of the models

Fig. 2. Correlation of the models

Fig.1 displays the responses of the models to the test face. The top panels present: 1a) a stimulus with additive random noise added, 1b) the original stimulus. The bottom panels show: 1c) the response of Model 1 (simple autoassociator), 1d) the response of Model 2 (Sobel operator), 1e) the response of Model 3 (Canny-Deriche filter) and 1f) the response of Model 4 (wavelet transform). The quality of recognition can be measured by computing the cosine between the vector $\widehat{\mathbf{x}}$ (i.e., the response of model) and $\mathbf{x}_k$ (i.e., the original stimulus which is also the desired response or target). Fig.2 shows the correlation between response and target for the 4 models used. Clearly, the standard method is the worst of all. Pre-processing the images improves the performance of the autoassociator, with the wavelet transform giving the best result. In conclusion, the multiscale resolution (i.e., wavelet pre-processing) approach leads to the best performance for the autoassociator. Therefore, we decided, in what follows, to consider only this approach.

3   Pattern completion of noisy patterns

We have applied the multiscale edge pre-processing to store a set of 40 Caucasian faces (20 males and 20 females). In order to evaluate further the improvement in

performance due to pre-processing, we have decided to test the model with different levels of Gaussian random noise added to the test stimulus. Learning was implemented as described for Model 1 (standard autoassociator) and Model 4 (wavelet). The procedure was the same as in the previous section except that the faces were tested with all levels of noise. Four levels of signal to noise ratio (expressed as the ratio of their respective variance) were selected: 1, 3/5, 3/8 and 3/13. Fig. 3 displays an example of the response of the models (standard and wavelet) as a function of the intensity of the noise added to a previously learned face used as test.
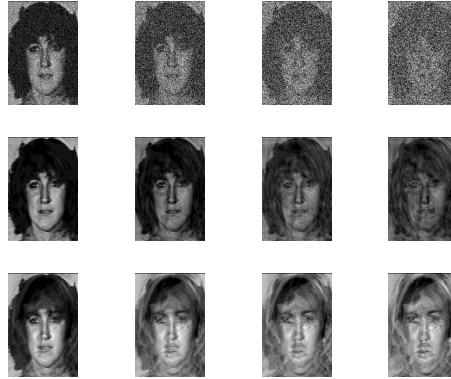
Fig. 3. *The top panels show 4 stimuli, the middle panels the responses produced by the autoassociator trained with the wavelet and the bottom panels the response of the autoassociator trained with the standard learning.*
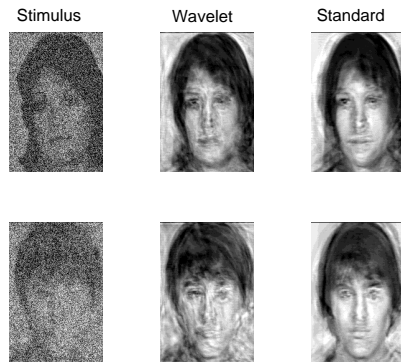
Fig. 4. Stimuli and responses.
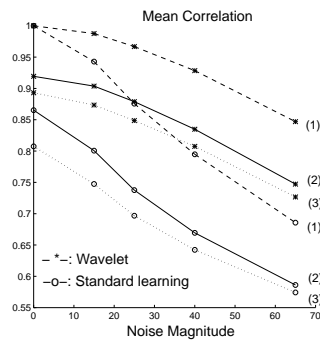
Fig. 5. Mean correlation functions.

We also decided to explore the performance of the model with 3 different types of face stimuli: 1) previously learned faces, 2) new faces similar to the learned faces, and 3) new faces coming from an other race than the learned faces. This was done in order to evaluate the robustness of the models in terms of response generalization to new stimuli. Fig. 4 displays, as an example, the responses of

both models for 2 new faces (from top to bottom): 1) a new face similar to the set of learned faces (Caucasian face), and 2) a new face face different from the set of learned faces (Japanese face). As can be seen in Fig. 4, better results are obtained with wavelet preprocessing. Fig. 5 displays the mean correlation between noiseless face images and the output for each model: 1) for 10 previously learned Caucasian faces , 2) for 10 new Caucasian faces, 3) for 10 new Japanese faces. In all cases, pre-processing the image improves the performance of autoassociator with the improvement being more important when the noise added is larger.

4   Conclusion

In this paper, we have explored the effects of storing, in a linear auto-associator, filtered versions of face images in addition to the original images. Compared to the Sobel operator and the simple Canny-Deriche operator the multiscale Canny-Deriche operator (i.e., a wavelet filter) gives the best performance for a pattern completion task involving degraded a degraded face image. The multiscale Canny-Deriche operator produces better generalization performance than the control with or without noise added to the image. The larger the amount of noise added, the larger the improvement in performance. We are now exploring the effects of using the multiscale Canny-Deriche operator for other traditional face processing tasks (see, e.g., [10]).

References

1. Kohonen, T.: *Associative memory: A system theoretic approach.* Springer-Verlag, Berlin. (1977)
2. Anderson, J.A., Silverstein, J.W., Ritz S.A. and Jones, R.S.: Distinctive features, categorical perception, and probability learning: Some applications of a neural model, *Psychological Review, 84,* 413–451. (1977)
3. Valentin, D., Abdi, H., O'Toole, A.J.: Categorization and identification of human face images by neural networks: A review of the linear autoassociative and principal component approaches. *Journal of Biological Systems, 2,* 413–429. (1994)
4. Valentin, D., Abdi, H., O'Toole, A.J. Cottrell, G.W.: Connectionist models of face processing: A survey. Pattern Recognition, 27, 1209–1230.
5. Abdi,H.: *Les Réseaux de neurones.* Presses Universitaires de Grenoble, Grenoble. (1994)
6. Jia,X. and Nixon,S.: Extending the feature vector for automatic face recognition. *IEEE-Transactions on Patterns Analysis and Machine intelligence, 17,* (1995)
7. Deriche, R.: Using canny's Criteria to Derive a Recursively Implemented optimal Edge Detector. *International Journal of Computer Vision, 1* (1987)
8. Bourennane, E., Paindavoine, M. and Truchetet, F.: Amélioration du filtre de Canny-Deriche pour la détection des contours sous forme de rampe. *Traitement du signal: Recherche, 10* (1993)
9. Mallat, S. and Zhong, Z.: characterization of signal from multiscale edges. IEEE-PAMI, 14, (1992)
10. Valentin, D., Abdi, H.: Can a linear autoassociator recognize faces from new orientations? *Journal of the Optical Society of America, series A, 13,* 717-724. (1996)

This article was processed using the LaTeX macro package with LLNCS style