16

Selection and Efficient Use of Local Features for Face and Facial Expression Recognition in a Cortical Architecture

Masakazu Matsugu
Canon Inc
Japan

1. Introduction

There are growing physiological and practical evidences that show usefulness of component (e.g., local feature) based approaches in generic object recognition (Matsugu & Cardon, 2004; Wolf et al., 2006; Mutch & Lowe, 2006; Serre et al., 2007) which is robust to variability in appearance due to occlusion and to changes in pose, size and illumination.

It is no doubt clear that low level features such as edges are important and utilized in most of visual recognition tasks. However, there are only a few studies that address economical and efficient use of intermediate visual features for higher level cognitive function (Torralba et al., 2004; Opelt et al., 2006). In this chapter, inspired by cortical processing, we will address the problem of efficient selection and economical use of visual features for face recognition (FR) as well as facial expression recognition (FER).

We demonstrate that by training our previously proposed (Matsugu et al., 2002) hierarchical neural network architecture (modified convolutional neural networks: MCoNN) for face detection (*FD*), higher order visual function such as FR and FER can be organized for shared use of such local features. The MCoNN is different from those previously proposed networks in that training is done layer by layer for intermediate as well as global features with resulting receptive field size of neurons being larger for higher layers. Higher level (e.g., more complex) features are defined in terms of spatial arrangement of lower level local features in a preceding layer. In the chapter, we will define a common framework for higher level cognitive function using the same network architecture (i.e., MCoNN) as substrate as follows.

- In Section 2, we will demonstrate two examples of *learning local features* suitable for *FD* in our MCoNN (Matsugu & Cardon, 2004). One approach is heuristic, supervised training by showing exemplar local features or patches of images, and the other is unsupervised training using SOM (self-organizing map) combined with supervised training in MCoNN.
- In the proposed framework, both FR and FER utilize common local features (e.g., corner like end-stop structures) learnt from exemplary image fragments (e.g., mouth corners, eye-corners) for *FD*. Specifically, in Section 3, spatial arrangement information of such local features is extracted implicitly for FR as feature vectors used in SVM classifiers (Matsugu et al., 2004). In the case of FER described in Section 4, spatial arrangement of

Source: Face Recognition, Book edited by: Kresimir Delac and Mislav Grgic, ISBN 978-3-902613-03-5, pp.558, I-Tech, Vienna, Austria, June 2007

common local features is used explicitly for rule-based analysis (Matsugu et al., 2003). We will show, by simulation, that learnt features for *FD* turn out to be useful for FR and FER as well.

2. Learning Local Features for Generic Object Detection

2.1 Modified convolutional neural network (MCoNN)

Convolutional neural networks (CoNN), with hierarchical feed-forward structure, consist of feature detecting (FD) layers, each of which followed with a feature pooling (FP) layer or sub-sampling layer. CoNN (LeCun and Bengio, 1995) as well as *Neocognitrons* (Fukushima, 1980) have been used for face detection (Matsugu et al., 2002; Osadchy et al., 2004) and recognition (Lawrence et al., 1995).

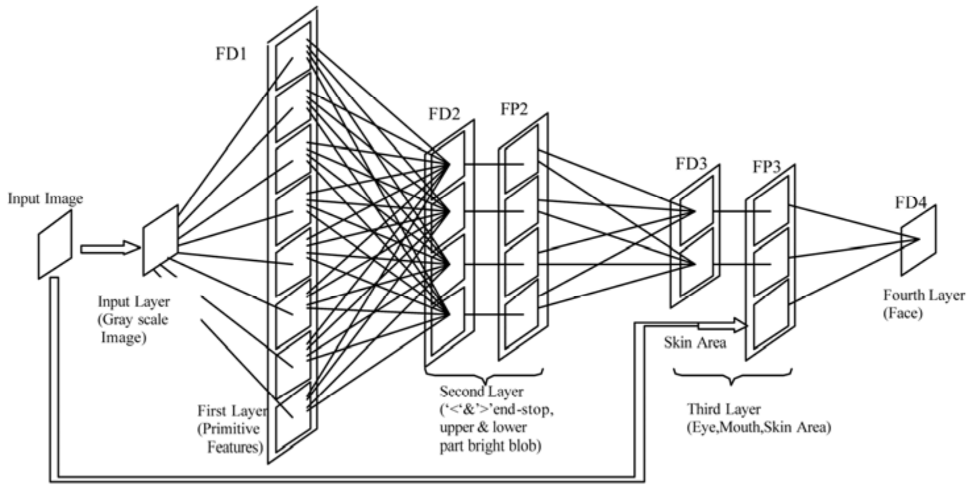

Figure 1. Modified convolutional neural network (MCoNN) architecture for facedetection

Proposed architecture in Figure 1 comes with the property of robustness in object recognition such as translation and deformation invariance as in well-known *neocognitrons*, which also have similar architecture. The MCoNN contains the same three properties as the original CoNN as well as Neocognitrons: local receptive fields, shared weights, and alternating feature detection/pooling mechanism to detect some intermediate (in the sense that local but not too simple) local features. Those properties are can be widely found in cortical structures (Serre et al., 2005). Feature pooling (FP) neurons perform either maximum value detection as in Riesenhuber & Poggio (1999) and Serre et al. (2007) or local averaging in their receptive fields of appropriate size.

Our model (MCoNN) for face detection as shown in Figure 1 is different from traditional ones in many aspects. First, it has only FD modules in the bottom and top layers. The intermediate features detected in FD2 constitute a set of figural alphabets (Matsugu et al., 2002; Matsugu & Cardon, 2004). Local features in FD1 are used as bases of figural alphabets, which are used for eye or mouth detection. Face detecting module in the top layer is fed with a set of outputs from facial component (e.g., such as eye, mouth) detectors as spatially ordered set of local features of intermediate complexity.

Second, we do not train FP (or sub-sampling) layers (FP neurons perform either maximum value detection or local averaging in their receptive fields). Third, we use a detection result of skin color area as input to the face detection module in FD4. The skin area is obtained simply by thresholding of hue data of input image in the range of [-0.078,0.255] for the full range of [-0.5,0.5], which is quite broad indicating that skin color feature plays merely auxiliary part in the proposed system.

Third, in our MCoNN model, in contrast to the original CoNN, local features to be detected in respective layers are pre-defined, and trained module by module (i.e., for each local feature class) for specifi category of local features; edge-like features in the first layer, and then in the second layer, corner-like structures (i.e., '<' and '>' end-stop), elongated blobs (i.e., upper part bright blob, and lower part bright blob) are detected. The second and third layers are composed of feature detecting layer and feature pooling layer as in original CoNN and Neocognitrons. Local features detected in the second layer constitute some alphabetical local features in our framework, and details will be explained in the next section. Eye and mouth features are detected in the third layer. Finally, a face is detected in the forth layer using outputs from the third layer and skin area data defined by some restricted range of hue and saturation values.

The training proceeds as follows. As in (Matsugu et al., 2002, Mitarai et al., 2003), training of the MCoNN is performed module by module using fragment images as positive data extracted from publicly available database (e.g., Softpia Japan) of more than 100 persons. Other irrelevant fragment images extracted from background images are used as negative samples. In the first step, two FD layers from the bottom, namely FD1 with 8 modules and FD2 with 4 modules, are trained using standard back-propagation with intermediate local features (e.g., eye corners) as positive training data sets. Negative examples that do not constitute the corresponding feature category are also used as false data. Specifically, we trained the FD2 layer, the second from the bottom FD layer to form detectors of intermediate features, such as end-stop structures or blobs (i.e., end-stop structures for left and right side and two types of horizontally elongated blobs (e.g., upper part bright, lower part bright) with varying sizes, rotation (up to 30 deg. with rotation in-plane axis as well as head axis). These features for training are fragments extracted from face images. More complex local feature detectors (e.g., eye, mouth detectors, but not restricted to these) are trained in the third or fourth FD layer using the patterns extracted from transforms as in the FD2 layer. As a result of these training sequences, the top FD layer, FD4, learns to locate faces in complex scenes. The size of partial images for the training is set so that only one class of specific local feature is contained. The number of training data set is 14847 including face images and background image for FD4 module, 5290 for FD3, and 2900 for FD2.

2.2 Supervised learning of local features as figural alphabets in MCoNN

Selecting optimal local features for multi-class object detection (Papageorgiou et al, 1998) is a crucial step toward generic object recognition. Face detection, face recognition, and facial expression recognition are no exceptions. In Burl et al. (1995) and Weber et al. (2000), an interest point operator and a k-means clustering algorithm are used to extract and regroup high-level features for estimating the parameters of the underlying probabilistic model.

In Ikeda et al. (2001), *image entropy* was adopted to select interesting areas in an image and also Self-Organizing Map (SOM) (Kohonen, 1985) was used to organize great amount of extracted high-level features, then a clustering algorithm was used to regroup similar units in the SOM to a certain number of macro-classes. In this section (Matsugu & Cardon, 2004), we explain sequential supervised training scheme to form a set of intermediate level feature detectors (Matsugu et al., 2002) and sub-optimal feature selection. For training the modified convolutional neural network (MCoNN), we extracted local image patches (Figure 2) around key points detected by Harris interest point operators.

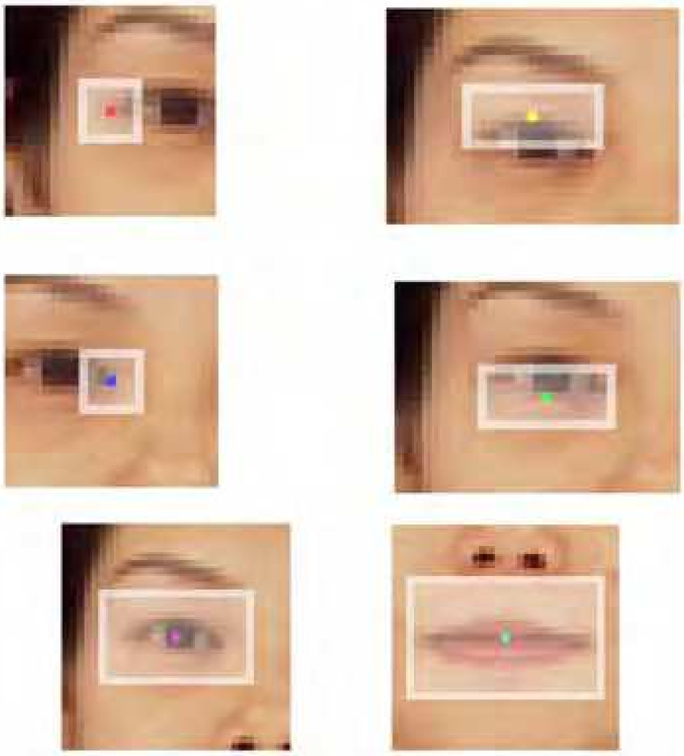

Figure 2. Local image fragments for training the second and third layers of MCoNN

Here a variant of back-propagation algorithm is used to train each layer separately (sequential BP: SBP) so that the extracted features are controlled, and also some specific parts of the face can be detected. The first two layers are trained with intermediate-level features (e.g. eye-corners), while the subsequent layers are trained with more complex, high-level features (e.g. eyes, faces...). This requires selecting a training set of features. By selecting a limited set of features for a specific object, we may expect to find a restricted yet useful set of receptive fields as in neurophysiological studies (Blackmore and Cooper, 1970; Hubel and Wiesel, 1962).

To find these features we apply classical BP (hereafter referred as GBP: global BP), not the proposed SBP, to the entire MCoNN with connections below Layer 2(FD1-FD2-FP2) in Figure 3 fixed, and analyze the output of Layer3 (high-level features). The GBP converges to a local minimum, therefore the algorithm will tend to extract sub-optimal features to minimize the detection error.

To examine the validity of our scheme of using MCoNN trained by GBP for generic object detection, we applied the MCoNN for face detection to the detection of bright-colored cars with significant variance in shape, illumination, size and orientation. The size of the images used for learning was 156 x 112, and 90 images were used for training and 10 images for validation. We aimed to find characteristic high-level features for the detection of this type of objects under particular view. In addition, it was necessary to tailor our model to be able to distinguish between cars and other rectangular objects. For this reason, we included a set of negative non-car examples, with similar rectangular shape but which were not cars.

2.3 Unsupervised learning of local features as figural alphabets in MCoNN

In this section (Matsugu & Cardon, 2004), we present an unsupervised feature extracting and clustering procedure, using an interest operator combined with a SOM. In contrast with Opelt et al. (2006), we do not use AdaBoost framework for this task. Instead, proposed method combines the advantages of both Weber et al. (2000) and Ikeda et al. (2001) by selecting a limited number of features and regrouping them using a topographic vector quantizer (SOM); acting like a vector quantizer and introducing a topographic relation at the same time. The obtained feature classes are self-organized, low-and intermediate-level features that are used to train the two first layers of the MCoNN and obtain a minimum set of alphabetical receptive fields.

Those alphabets as in Opelt et al. (2006) considerably reduces the complexity of the network by decreasing the number of parameters and can be used for detection of different object classes (e.g. faces, cars,...). We also introduce a method to select optimal high-level features and illustrate it with the car detection problem.

The whole network for face detection as well as car detection is described in the lower part of Figure 3. Some specific local fragments of image extracted a priori, by using the proposed method in this study, are used to train the first two layers of the MCoNN. First, we train the MCoNN to recognize only one feature (one output plane in FD2). A sequential back-propagation algorithm (Matsugu et al., 2002) is used for learning and weights are updated after each training pattern (fragments of images) is presented. A fixed number of 100 epochs has been used. For each training set, a different number of cell-planes in layer S1 have been tested. The network has essentially four distinct sets of layers: FD1, FD2-FP2, FD3-FP3, FD4 (FD_k: the *k*th feature detecting layer; FP_j: the *j*th feature pooling layer for subsampling). Layers FD3-FP3 and above are concerned with object specific feature detection. In order to limit the number of features to object-relevant features, an interest point operator is used. This operator selects corner-like features in the image.

Having selected a restricted number of points using keypoint detector (Harris & Stephens, 1988; Lowe, 1999;Kadir & Brady, 2001; Csurka et al. 2004) we extract features around these points. These features are used as learning set for the SOM well suited for classifying and visualizing our feature set. It turned out that the illumination has a big influence on the classification of our features, so we have rescaled the feature set between -1 and 1 before applying the SOM. Each unit of the SOM defines a training set for the MCoNN.

Once lower-level alphabetical feature detectors are formed, higher level feature detectors can be obtained from BP with connections between neurons below intermediate layers fixed. Since we are interested in low-level features to train the first two layers of the MCoNN, we have chosen to extract small features as shown in Figure 3 (upper left). After applying SOM with those fragment images extracted from a database of 904 (size: 208 x 256) images (300 faces (frontal view), 304 cars (upper view) and 300 various types of images), we obtained a set of 69,753 features. From these features, we manually selected some prototypical features that have simple characteristics (e.g., horizontal, vertical, and diagonal contrast).

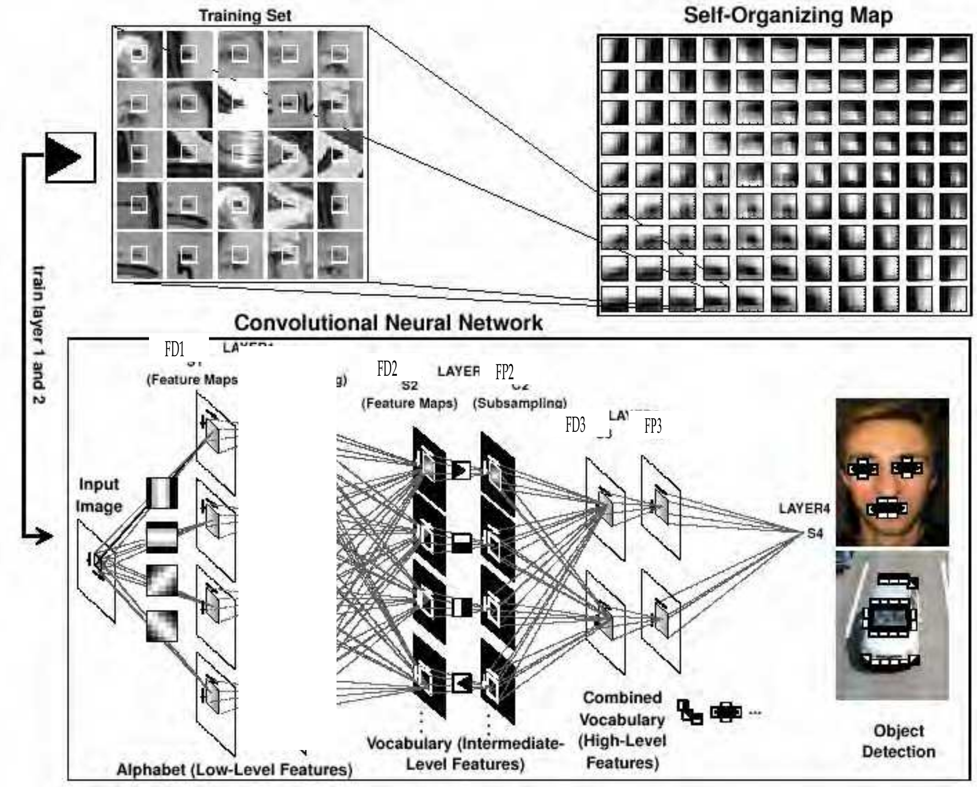

Figure 3. Schematic diagram of learning system for generic object recognition (adapted from Matsugu & Cardon, 2004)

We used the SOM Toolbox in Matlab and fixed the number of units to 100 based on the assumption that there are not more than 100 different types of local features (figural alphabets) for generic object detection. Fragmented image patches for clustering are appropriately cropped so that irrelevant background features are cut out.

For each cluster we only consider the 300 features, which are the closest to the SOM-unit, in terms of Euclidean distance. 200 features are used for training, 50 features for validation and the last 50 units for testing. The results have been obtained with a test set of 50 features and optimal receptive fields have been selected by cross-validation. We see that for such simple features, only one cell-plane in S1 is sufficient to obtain good detection results. We also notice that the learnt receptive fields (Figure 4) have a regular pattern. Based upon these patterns we use a set of four alphabetical patterns ***V***, ***H***, ***S***, ***B*** (hereafter, represents vertical, horizontal, slash, backslash, respectively) described in Figure 4.

We observe that some feature clusters in the SOM have a more complex aspect as shown in Figure 3. We claim that these more complex features can be detected using the simple receptive fields, described in the previous section. Considering for example the feature described in Figure 3, we see that this eye-corner type feature can be decomposed into two local alphabetical features (Figure 5).

The usefulness of our alphabetical set appears when we want to detect, using a small number of receptive fields, a bit more higher-level features with more complex geometrical or textual structures. Let us consider the features used to detect a complete eye or a mouth (Matsugu et al., 2002). They can be decomposed to two horizontal, two slash and two backslash components (Figure 6).

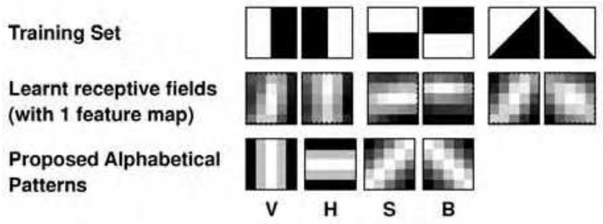

Figure 4. Alphabetical patterns obtained from SOM which are used for training McoNN. Resuling receptive fields of McoNN correspond to each feature detector

With a limited set of three fixed receptive fields ***H***, ***S*** and ***B*** it turned out that we reach a detection rate of eye-corner comparable to that of using six learnt receptive fields. Our alphabetical set, being close to the optimal set of weights, therefore outperforms the learnt weights. We can extend these results for different types of complex features and construct a vocabulary set that can be recognized with ***H***, ***V***, ***S***, and ***B***. For illustration purposes, we have tested our alphabet with images from which features have been extracted. It turned out that we could detect, in the S2 layer, eye- and mouth-corners as well as the side mirrors of a car, using only three receptive fields (***H***, ***S*** and ***B***).

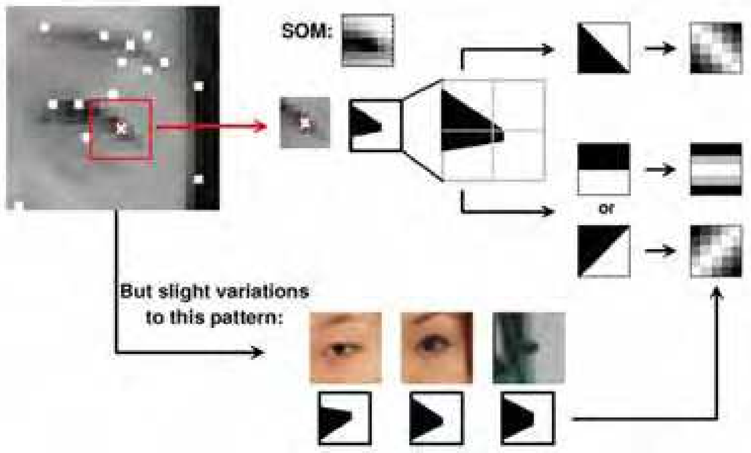

Figure 5. Example (corner-like structure) of local feature extracted from local image patches of eye as a figural alphabet and its decomposition into three elementary features

An interesting question to be answered is which vocabulary we should use, in other words, what features are important to detect a specific object. To find these features we apply classical BP (hereafter referred as GBP: global BP), not the proposed SBP, to the entire MCoNN with connections below S3 layer (FD1--FD2-FP2) fixed, and analyze the output of Layer3 (high-level features). The GBP converges to a local minimum, therefore the algorithm will tend to extract sub-optimal features to minimize the detection error.

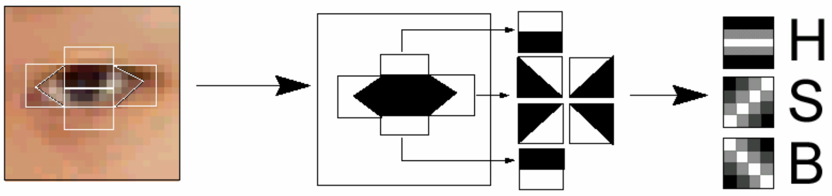

Figure 6. Example of visual vocabulary that constitues *eye* as a constellation of figural alphabet in the proposed system

Having discovered the important features for our object detection problem, we obtain object specific vocabulary to select to construct these high-level features. We can use SBP as in (Matsugu et al., 2002) to train the higher level layers in the MCoNN: to train layer by layer with the selected vocabulary features.

In spite of the simplicity of this alphabet it gives remarkable results, comparable and sometimes better than the learnt receptive fields with average detection rate over 95% for different types of features. After obtaining alphabetical feature detectors in the S1 and S2 layer of MCoNN, we applied GBP to the S3 and S4 layers of MCoNN, with lower level weights fixed, to obtain higher level feature detectors (e.g., cars and faces), thereby obtaining sub-optimal vocabulary set. The optimality was examined in terms of cross-validation.

3. Component-based Face Recognition

3.1 Literature overview

Face recognition algorithms have been extensively explored (Belhumeur et al., 1997; Brunelli & Poggio, 1993; Turk & Pentland, 1997; Guodong et al., 2000; Heisele et al., 2001; Heisele & Koshizen, 2004; Li et al., 2000; Moghaddam et al., 1998; Pontil & Verri, 1998; Wiskott et al., 1997) and most of which address the problem separately from object detection, which is associated with image segmentation, and many assume the existence of objects to be recognized without background. Some approaches, in the domain of high-level object recognition, address economical use of visual features extracted in the early stage for object detection. However, only a few object recognition algorithms proposed so far explored efficiency in the combined use of object detection and recognition (Li et al., 2000).

For example, in the dynamic link matching (DLM) (Wiskott et al., 1997), Gabor wavelet coefficient features are used in face recognition and detection as well. However, we cannot extract shape as well as spatial arrangement information on facial components directly from those features since, for a set of nodes of the elastic graph, they do not contain such information. This necessitated to device the graph matching technique, a computationally expensive procedure, which requires quite different processing from feature detection stage.

Convolutional neural networks (CoNN) (Le Cun & Bengio, 1995) have been exploited in face recognition and hand-written character recognition. In (Matsugu et al., 2001, 2002), we proposed a MCoNN model for robust face detection. SVM has also been used for face recognition (Guodong et al., 2000; Heisele et al., 2001; Heisele & Koshizen, 2004; Li et al., 2000; Pontil & Verri, 1998). In particular, in (Heisele et al., 2001; Heisele & Koshizen, 2004), SVM classification was used for face recognition in the component-based approach.

This section, in the domain of face recognition as a case study for general object recognition with object detection, explores the direct use of intermediate as well as low level features obtained in the process of face detection. Specifically, we explore the combined use of our MCoNN and support vector machines (SVM), the former used for feature vector generation, the latter for classification. Proposed algorithm is one of component-based approaches (Heisele et al., 2001; Heisele & Koshizen, 2004) with appearance models represented by a set of local, area-based features. The direct use of intermediate feature distributions obtained in face detection, for face recognition, brings unified and economical process that involves simple weighted summation of signals, implemented both in face detection and recognition.

3.2 Proposed component based face recognition

Proposed face recognition system (Matsugu et al., 2004) utilizes intermediate features extracted from face detection system using MCoNN, which are fed to SVM for classification. This combination of MCoNN with SVM is similar in spirit to recent works by Serre et al. (2007) and Mutch & Lowe (2006). Figure 7 shows detailed structure of the MCoNN for face detection as well as face recognition. Here, we describe feature vectors and the procedure for their generation in face recognition. A feature vector, $\boldsymbol{F}$, used in SVM for face recognition is an N dimensional vector, synthesized from a set of local output distributions, $\boldsymbol{F}_1$ (as shown in Figure 2(1)), in a module detecting edge-like feature in FD1 layer in addition to output distributions, $\boldsymbol{F}_2$, (as shown in Figure 2(2)) of two intermediate-level modules detecting eye and mouth in FD2 layer. Thus, $\boldsymbol{F} = (\boldsymbol{F}_1, \boldsymbol{F}_2)$ where $\boldsymbol{F}_1 = (\boldsymbol{F}_{11}, \ldots, \boldsymbol{F}_{1m})$ and $\boldsymbol{F}_2 = (\boldsymbol{F}_{21}, \ldots, \boldsymbol{F}_{2n})$ are synthesized vectors formed by component vectors, $\boldsymbol{F}_{1k}$ (k=1, ..., m) and $\boldsymbol{F}_{2k}$ (k=1, ..., n), respectively.

Each component vector represents possibility or presence of specific class of local feature in an assigned local area. Dimension of a component vector is the area of a rectangular region as in Figure 9. Thus dimension of feature vector, N, is the total summation of respective dimensions of component vectors. In particular, $\boldsymbol{F}_1 =(\boldsymbol{F}_{11}, \boldsymbol{F}_{12}, \ldots, \boldsymbol{F}_{1,15})$, and local areas, total number of assigned areas being 15 as in Figure 9 (1), for component vectors are set around eye, nose, and mouth, using the detected eye location from the MCoNN. $\boldsymbol{F}_1$ reflects shape information of eye, mouth, and nose. $\boldsymbol{F}_2 = (\boldsymbol{F}_{21}, \boldsymbol{F}_{22}, \boldsymbol{F}_{23})$, and each component vector reflects spatial arrangement of eye or eye and nose, etc., depending on how local areas in FD2 (e.g., positions and size) are set.

The procedure for feature vector generation is summarized as follows. First, we define a set of local areas for FD1 as well as FD3 modules based on the CNN output in FD3 modules for eye and mouth detection. Positions of local areas in FD1 module are set around specific facial components (i.e., eyes, mouth) as illustrated in Figure 9 (1). The size of respective local areas in the output plane of FD1 module is set relatively small (e.g., 11 x 11) so that local shape information of figural alphabets can be retained in the output distribution, while the local area in the FD2 plane is relatively larger (e.g., 125 x 65) so that information concerning spatial arrangement of facial components (e.g., eye) is reflected in the distribution of FD2 outputs.

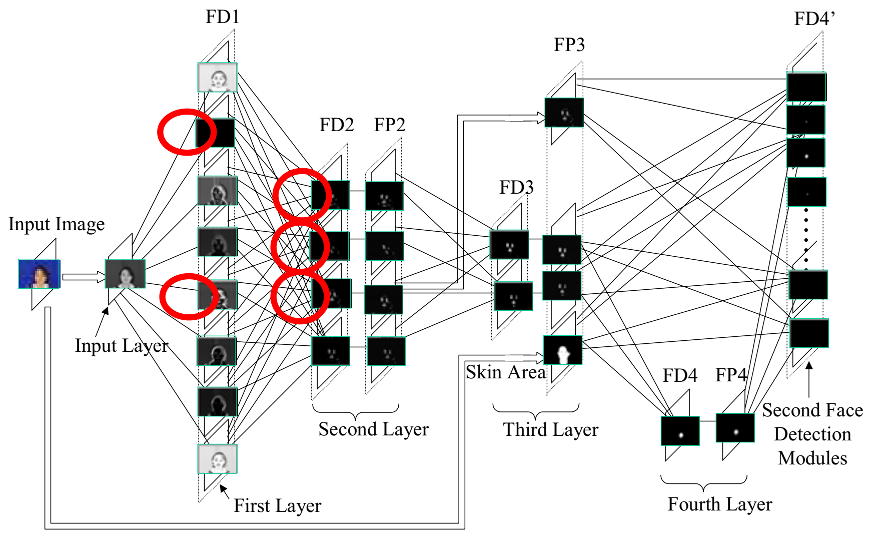

Figure 7. MCoNN for face recognition and facial expression recognition. Outputs from encircled modules in FD1 and FD2 layers are used for face recognition

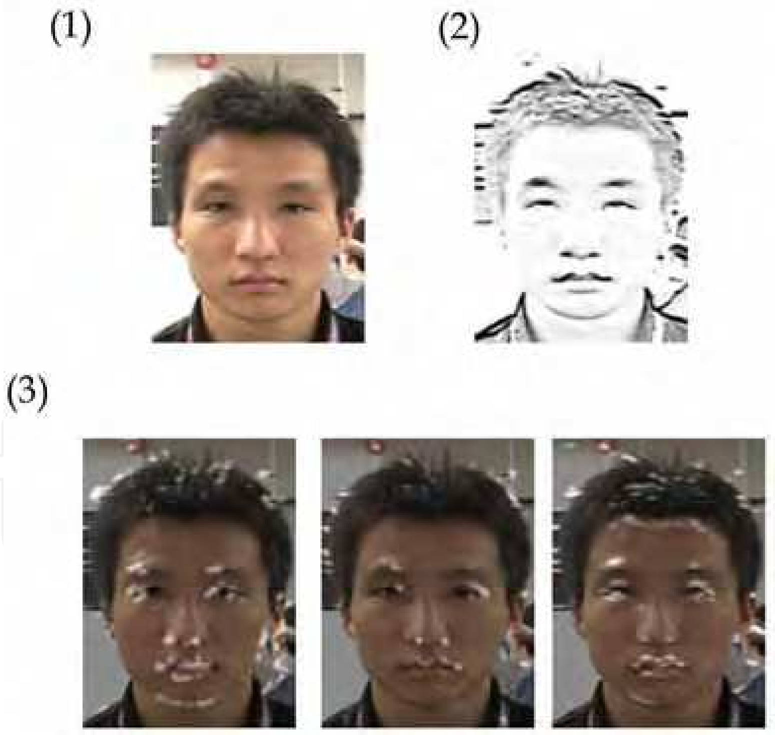

Figure 8. Intermediate output from MCoNN (1):input image, (2) output example from FD1, (3) intermediate outputs from encircled modules of FD2 in Figure 7

For face recognition, we use an array of linear SVMs for one-against-one multi-class recognition of faces. The SVM library used in the simulation is *libsvm2.5*, available in the public domain. In the SVM training, we used a dataset of FVs extracted for each person in the way described in Section3.

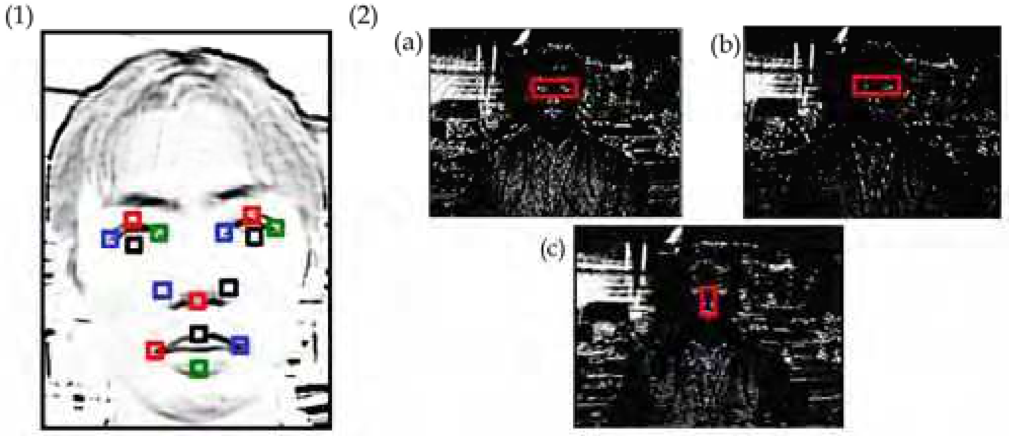

Figure 9. Local areas for face recognition. (1): small local area for local shape description, (2): mid-level local area for mid-level description of intermediate local feature configuration. (a,b,c): outputs from '< end-stop', '> end-stop', 'upper part bright horizontal blob' detectors, respectively

The size of input image is of VGA, and the size of local areas for FVs is 15 x15, 125 x 65, or 45 x 65 depending on the class of local features. As indicated in Figure 9 (1), the number of local areas for FD1 feature and FD2 feature is fourteen and two, respectively. The number of FVs for one person is 30, which are obtained under varying image capturing conditions so that size, pose, facial expression, and lightning conditions of respective faces are slightly different.

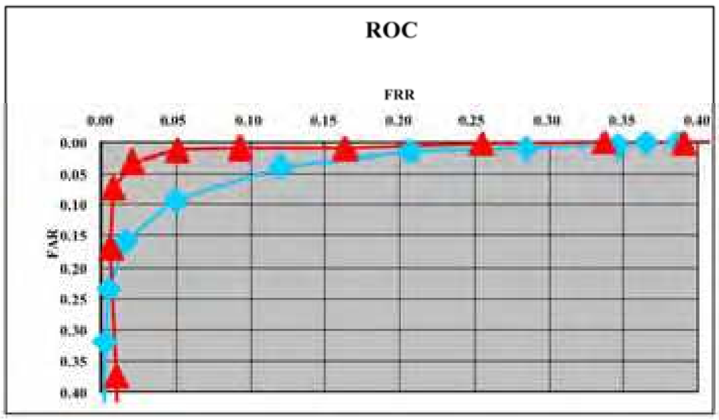

Figure 10. ROC curve of face recognition for 20 people. Triangle-red curve: ROC from intermediate outputs from MCoNN, diamond-blue curve: ROC obtained from raw input data fed to SVM

Face image database used ofr training and testing is in-house DB (10 subjects, 1500 images) and PIE database (we used part of the DB: 15 subjects 60 images) by CMU. We compared results obtained from McoNN's intermediate outputs with those obtained from raw data using the same local area as in Figure 9. ROC curves in Figure 10 obtained for in-house face database show that using intermediate outputs rather than raw data provide better performance. Using the same dataset, we compared our model with commercially available software which is based on DLM (Wiskott et al., 1997). The recognition rate turned out to be almost the same for the relative size of 0.8 to 1.2, while F.A.R. is slightly inferior to our model (i.e., F.A.R. is not perfectly zero), suggesting that our model involving much simpler operations equals to the performance of one of the best models (Matsugu et al., 2004).

4. Component-based Facial Expression Recognition

4.1 Literature overview

Facial expressions as manifestations of emotional states, in general, tend to be different among individuals. For example, smiling face as it appears may have different emotional implications for different persons in that 'smiling face', perceived by others, for some person does not necessarily represent truly smiling state for that person. Only a few algorithms (e.g., Ebine & Nakamura, 1999) have addressed robustness to such individuality in facial expression recognition. Furthermore, in order for facial expression recognition (FER) to be used for human-computer-interaction, for example, that algorithm must have good ability in dealing with variability of facial appearance (e.g., pose, size, and translation invariance).

Most algorithms, so far, have addressed only a part of these problems (Wallis & Rolls, 1997). In this study, we propose a system for facial expression recognition that is robust to variability that originates from individuality and viewing conditions. Recognizing facial expression under rigid head movements was addressed by (Black & Yacoob, 1995). Neural network model that learns to recognize facial expressions from an optical flow field was reported in (Rosenblum et al., 1996). Rule-based system was reported in (Yacoob & Davis, 1996) and (Black & Yacoob, 1997), in which primary facial features were tracked throughout the image sequence. Recently, Fasel (2002) has proposed a model with two independent convolutional neural networks, one for facial expression and the other for face identity recognition, which are combined by an MLP.

4.2 Facial expression recognition using local features extracted by MCoNN

We show, in this section, proposed rule-based processing scheme to enhance subject independence in facial expression recognition. We found that some of lower level features extracted by the first FD layer of MCoNN for face detection as well as face recognition are also useful for facial expression recognition. Primary features used in our model are horizontal line segments made up of edge-like structures similar to step and roof edges (extracted by two modules in FD1 layer, circled in Figure 7 representing parts of eyes, mouth, and eyebrows. For example, changes in distance between end-stops (e.g., left-corner of left eye and left side end-stop of mouth) within facial components and changes in width of line segments in lower part of eyes or cheeks are detected to obtain saliency scores of a specific facial expression. Primary cues related to facial actions adopted in our facial analysis for the detection of smiling/laughing faces are as follows.

1. Distance between endpoints of eye and mouth gets *shorter* (lip being raised)

2. Length of horizontal line segment in mouth gets *longer* (lip being stretched)
3. Length of line segments in eye gets *longer* (wrinkle around the tail of eye gets longer)
4. Gradient of line segment connecting the mid point and endpoint of mouth gets *steeper* (lip being raised)
5. Step-edge or brightness inside mouth area gets *increased* (teeth being appeared)
6. Strength of edges in cheeks *increased* (wrinkle around cheeks being grown)

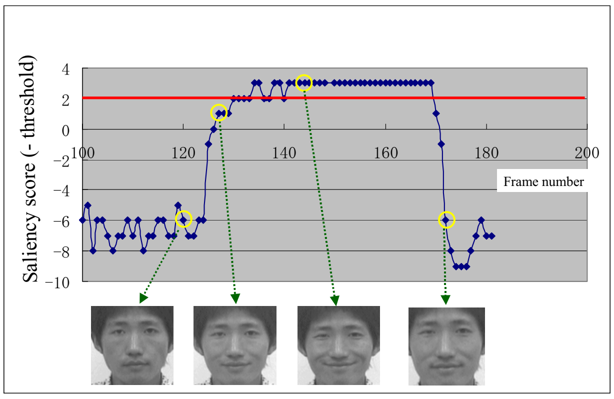

Figure 11. Normalized saliency score subtracted by constant value for smiling face detection

We use these multiple cues as supporting evidence of specific facial expression (i.e., smile). Each cue was scored based on the degree of positive changes (i.e., designated changes as given above) to the emotional state (e.g., happiness). Saliency score of specific emotional state is calculated with weighted summation of respective scores, which is then thresholded for judging whether the subject is smiling/laughing or not. Greater weighting factors are given to cues of less individuality (i.e., more common cues across individuals): (i), (ii), and (v). Figure 11 shows a sequence of normalized saliency scores indicating successful detection of smiling faces with an appropriate threshold level. The network demonstrated the ability to discriminate smiling from talking based on the duration of saliency score above threshold (longer duration implies greater possibility of *smiling*; Matsugu et al., 2004). We obtained results demonstrating reliable detection of smiles with recognition rate of 97.6% for 5600 still images of more than 10 subjects.

In contrast to a number of approaches (Donato et al., 1999), invariance properties in terms of translation, scale, and pose, inherent in our non-spiking version of MCoNN (Matsugu et al., 2002), brings robustness to dynamical changes both in head movements and in facial expressions without requiring explicit estimation of motion parameters. Because of the topographic property of our network which preserves the position information of facial features from bottom to top layers, the translation invariance in facial expression recognition is thus inherently built into our convolutional architecture with feedback mechanism for locating facial features.

Specifically, intermediate facial features such as eyes and mouth are detected and utilized for tracking useful primitive local features extracted by the bottom layer FD1 of MCoNN. Implicit location information of eyes and mouth detected in the MCoNN are used, through the feedback loop from the intermediate layer FP3, to confine the processing area of rule-based facial feature analysis, which analyzes differences in terms of at least six cues.

It turned out that the system is quite insensitive to individuality of facial expressions with the help of the proposed rule-based processing using single but individual normal face. Because of the voting of scores for various cues in terms of differences of facial features in neutral and emotional states, individuality is averaged out to obtain subject independence.

5. Conclusion

In this chapter, we reviewed our previously proposed leaning methods (unsupervised and supervised) for appropriate and shared (economical) local feature selection and extraction for generic face related recognition. In particular, we demonstrated feasibility of our hierarchical, component based visual pattern recognition model, MCoNN, as an implicit constellation model in terms of convolutional operation of local feature, providing a substrate for generic object detection/recognition. Detailed simulation study showed that we can realize face recognition as well as facial expression recognition efficiently and economically with satisfactory performances by using the same set of local features extracted from the MCoNN for face detection.

6. Acknowledgement

We used a face database HOIP by *Softpia Japan* to train the network for face detection.

7. References

Belhumeur, P., Hesoanha, P. & Kriegman, D. (1997). Eigenfaces vs fisherfaces: recognition using class specific linear projection, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, 711-720

Black, M. & Yacoob, Y. (1995). Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion, *Proc. IEEE Fifth Int. Conf. on Computer Vision*, 374-381

Black, M. & Yacoob, Y. (1997). Recognizing Facial Expressions in Image Sequences Using Local Parameterized Models of Image Motion, *Int. J. of Computer Vision*, vol. 25, 23-48

Blackmore, C. & Cooper, G. E. (1970). Development of the brain depends on the visual environment, *Nature*, vol. 228, 477-478

Brunelli, R. & Poggio T. (1993). Face recognition: features versus templates, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, 1042-1052

Burl, M., Leung, T. & Perona, P. (1995). Face localization via shape statistics, *Proc. Intl. Workshop on Automatic Face and Gesture Recognition*

Csurka, G., Dance, C. R., Fan, L., Willamowski, J. & Bray, C. (2004). Visual categorization with bags of keypoints, *Proc. European Conf. On Computer Vision*, Springer-Verlag, Berlin

Donato, G., Bartlett, M. S., Hager, J. C., Ekman, P. & Sejnowski, T. (1999). Classifying facial actions, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.21, 974-989

Ebine, H. & Nakamura, O. (1999). The recognition of facial expressions considering the difference between individuality (in Japanese), *Trans. IEE of Japan*, vol.119-C, 474-481

Fasel, B. (2002). Robust face analysis using convolutional neural networks, *Proc. Int. Conf. on Pattern Recognition*

Fergus, R., Perona, P. & Zisserman, A. (2003). Object class recognition by unsupervised scale-invariant learning, *Proc. IEEE Int. Conf. On Computer Vision and Pattern Recognition*

Földiák, P. (1991). Learning invariance from transformation sequences, *Neural Comput*. vol. 3, 194-200

Fukushima, K. (1980). Neocognitron: a self-organizing neural networks for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics* vol. 36, 193-202

Guodong, G., Li, S. & Kapluk, C. (2000). Face recognition by support vector machines. *Proc. IEEE International Conf. On Automatic Face and Gesture Recognition*, 196-201

Harris, C. & Stephens, M. (1988). A combined corner and edge detector, *Proc. Alvey Vision Conf.* 147-151

Heisele, B., Ho, P. & Poggio, T. (2001). Face recognition with support vector machines: global versus component-based approach. *Proc. International Conf. on Computer Vision,* 688-694

Heisele, B. & Koshizen, T. (2004). Components for face recognition *Proc. IEEE International Conf. on Automatic Face and Gesture Recognition*

Hubel D. & Wiesel T. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology* vol. 160**,** 106-154

Ikeda, H., Kashimura, H., Kato, N. & Shimizu, M. (2001). A novel autonomous feature clustering model for image recognition, *Proc. of the 8th International Conference on Neural Information Processing*

Kadir, T. & Brady, M. (2001). Scale, saliency and image description, *International Journal of Computer Vision*, vol. 45, 83-105

Kohonen, T. (1985). *Self-Organizing Maps*. Springer-Verlag, Berlin

Lawrence, S., Giles, G. L., Tsoi, A. C. & Back, A. D. (1995). Face recognition: a convolutional neural network approach, *IEEE Transactions on Neural Networks*, vol. 8, 98-113

Le Cun, Y. & Bengio, T. (1995). Convolutional networks for images, speech, and time series, In: Arbib, M.A. (ed.): *The handbook of brain theory and neural networks*, MIT Press, Cambridge, 255-258

Li, Y., Gong, S. & Liddel, H. (2000). Support vector regression and classification based multi-view face detection and recognition, *Proc. IEEE International Conf. on Automatic Face and Gesture Recognition*, 300-305

Lowe, D. (1999). Object recognition from local scale-invariant features, *Proc. IEEE international Conf. On Computer Vision*, 1150-1157

Matsugu, M. (2001). Hierarchical Pulse-coupled neural network model with temporal coding and emergent feature binding mechanism, *Proc. International Joint Conf. on Neural Networks (IJCNN 2001)*, 802-807

Matsugu, M., Mori, K., Ishii, M. & Mitarai, Y. (2002). Convolutional spiking neural network model for robust face detection, *Proc. International Conf. on Neural Information Processing (ICONIP 2002)*, 660-664

Matsugu, M., Mori, K., Mitarai, Y. & Kaneda, Y. (2003). Subject independent facial expression recognition with robust face detection, *Neural Networks*, vol. 16, 555-559

Matsugu, M. & Cardon , P. (2004) Unsupervised Feature Selection for Multi-class Object Detection Using Convolutional Neural Networks, *Advances in Neural Networks-ISNN 2004, LNCS 3173*, Springer-Verlag, Berlin, I-864-869

Matsugu, M., Mori, K. & Suzuki, T. (2004). Face recognition using SVM combined with CNN for face detection, *Proc. International Conf. On Neural Information Processing (ICONIP 2004), LNCS 3316*, 356-361, Springer-Verlag, Berlin

Mitarai, Y.,Mori, K.& Matsugu, M. (2003). Robust face detection system based on convolutional neural networks using selective activation of modules (In Japanese), *Proc. Forum in Information Technology*, 191-193

Moghaddam, B., Wahid, W. & Pentland, A. (1998). Beyond eigenfaces: probabilistic matching for face recognition, *Proc. IEEE International Conf. on Automatic Face and Gesture Recognition*, 30-35

Mutch, J. & Lowe, D. G. (2006). Multiclass object recognition with sparse, localized features, *Proc. IEEE Conf. On Computer Vision and Pattern Recognition*

Opelt, A., Pinz, A. & Zisserman, A. (2006). Incremental learning of object detectors using a visual shape alphabet, *Proc. IEEE Conf. On Computer Vision and Pattern Recognition*

Osadchy, M., Miller, M.L. & Le Cun, Y. (2004). Synergetic face detection and pose estimation with energy-based models, *Neural Information Processing*

Papageorgiou, C. P., Oren, M. & Poggio, T. (1998). A general framework of object detection, *Proc. IEEE International Conference on Computer Vision*, 555-562

Pontil, M. & Verri, A. (1998). Support vector machines for 3-d object recognition, *IEEE Trans.on Pattern Analysis and Machine Intelligence*, vol. 20**,** 637-646

Riesenhuber, M. & Poggio, T. (1999). Hierarchicaal models of object recognition in cortex, *Nature Neuroscience*, vol. 2, 1019-1025

Rosenblum, M., Yacoob, Y. & Davis, L.S. (1996). Human expression recognition from motion using a radial basis function network architecture, *IEEE Trans. Neural Networks*, vol. 7, 1121-1138

Serre, T., Kouch, M., Cadieu, C., Knoblich, U., Kreiman, G. & Poggio, T. (2005). A theory of object recognition: computations and circuits in the feedforward path of the ventral stream in primate visual cortex, *CBCL Memo*, 259, MIT, Cambridge

Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M. & Poggio, T. (2007). Robust object recognition with cortex-like mechanisms, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, 411-426

Torralba, A., Murphy, K.P. & Freeman, W. T. (2004). Sharing features: efficient boosting procedures for multiclass object detection, *Proc. IEEE Conf. On Computer Vision and Pattern Recognition*

Turk, M. & Pentland, A. (1991). Face recognition using eigenfaces, *Proc. IEEE Conf. On Computer Vision and Pattern Recognition,* 586-591

Wallis, G. & Rolls, E.T. (1997). Invariant face and object recognition in the visual system, *Prog. in Neurobiol*. vol. 51, 167-194

Weber, M., Welling, M. & Perona, P. (2000). Unsupervised learning of models for recognition, *Proc. of the 6th European Conference on Computer Vision*

Wiskott, L., Fellous, J.-M., Krüger, N. & von der Malsburg, C. (1997). Face recognition by elastic bunch graph matching. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, 775-779

Wolf, L., Bileschi, S. & Meyers, E. (2006). Perception strategies in hierarchical vision systems, *Proc. IEEE Conf. On Computer Vision and Pattern Recognition*

Yacoob, Y. & Davis, L. S. (1996). Recognizing human facial expression from long image sequences using optical flow, *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol.18, 636-642