LEARNING REPRESENTATIVE LOCAL FEATURES FOR FACE DETECTION

Xiangrong Chen, Lie Gu, Stan Z Li, Hong-Jiang Zhang

Microsoft Research China, Beijing, China, 100080
E-mail: liegu@microsoft.com

ABSTRACT

This paper describes a face detection approach via learning local features. The key idea is that local features, being manifested by a collection of pixels in a local region, are learnt from the training set instead of arbitrarily defined. The learning procedure consists of two steps. First, a modified version of NMF (Non-negative Matrix Factorization), namely local NMF (LNMF), is applied to get an overcomplete set of local features. Second, a learning algorithm based on AdaBoost is used to select a small number of local features and yields extremely efficient classifiers. Experiments are presented which show that the face detection performance is comparable to the state-of-the-art face detection systems.

1. INTRODUCTION

There is psychological [10] and physiological [19, 9] evidence for parts based representations in the brain. Some face detection algorithms also rely on such representations. However, the spatial shape of their local features is often subjectively defined instead of being learnt from the training data set.

Yang *et al*. [20] describe a method for frontal face detection on 20x20 regions. They assign a weight to every possible pixel value at every possible location within the region. The weights are determined by an iterative training procedure using the Winnow update rule. Once they have determined the weights they can classify any region by looking up and summing the weights corresponding to each pixel value. Thus each of their local features relies on only one pixel.

Colmenarez and Huang [5] used first order Markov Chain model over 11x11 input region to model face and non-face class conditional probabilities. To build the model, they calculate 1st order conditional probabilities for all pixels pairs, indicating that each of their local feature involve two pixels. The training procedure finds the mapping from the region into a 1 dimensional array with maximum sum of the corresponding 1st order conditional probabilities according to the training set. Any region can then be classified as face or non-face by looking up and summing the probabilities corresponding to the intensity values of each selected pixel pair.

Schneiderman and Kanade [16] argued that local features which are too small – one pixel at the extreme – will not be powerful enough to describe anything distinctive about the object. They use multiple appearance-based detectors that span a range of the object's orientation. Each detector uses a statistical model to represent object's appearances over a small range of views, to capture variation that cannot be modeled explicitly. They use rectangular subregions at multi-scales as local features in the statistical model. Size of those rectangles is pre-defined.

Burl and Perona [3] detected 5 types of features on the face: the left eye, right eye, nose/lip junction, left nostril, and right nostril. They assume that the feature detectors for each feature are fallible. Since they assume only one face is present in each image, at most one feature response is correct for each type of detector. Such hand-picked local features can also be found in Pentland's method [13], etc.

Rowley *et al*. [14] used a multilayer perceptron neural network system for classification. A 20x20 input region is divided into blocks of either 5x5, 10x10, or 20x5. Each hidden unit has one block as its receptive field. In their experiments with modular systems, they separately trained two or three of the above networks and then applied various methods for merging their results. Since the hidden units have only local support, we can infer that this particular network topology emphasizes local features over global one.

Viola and Jones [18] argued that the most common reason for using features rather than the pixels directly is that features can act to encode ad-hoc domain knowledge that is difficult to learn using a finite quantity of training data. Given a 24x24 region, they use an exhaustive set of three kinds of Harr like rectangular features. A following AdaBoost procedure is applied to learn important features from the overcomplete feature set. In contrast to their

method, Papageorgiou et al. [11] use a overcomplete set of Quadruple density 2D Harr basis at scales 4×4 and 2×2 pixels since they think the dimensions correspond to typical facial features for their 19×19 face images. they average the normalized coefficients over the entire set of example to identify the important Harr basis.

From the methods above we can conclude that there are two main steps for learning local features. The first step determines various characteristics of the local feature, including size, shape, location and calculations over the corresponding pixels, etc. Generally a overcomplete feature set is required for further selection of the features. The second step aims to find out the important features among the overcomplete set with the knowledge contained in the training data. Most previous face detection algorithms put learning procedure in the second step while little or no attention was put in the much, if not more, important first step. Instead, they define the spatial shape and other properties of their local features manually and intuitively.

Several existing algorithms can be applied to learn parts-based representation from examples. Local feature analysis (LFA) [12] is a method for extracting local topographic representation in terms of local features. The extraction is from the global PCA basis, also based on second order statistics. The LFA representation enables use of specific local features for identification instead of a global representation.

Independent component analysis [7, 6] is a linear non-orthogonal transform which makes unknown linear mixtures of multi-dimensional random variables as statistically independent as possible. It not only decorrelates the second order statistics but also reduces higher-order statistical dependencies. It extracts independent components even if their magnitudes are small whereas PCA extracts components having largest magnitudes. It is found that independent component of natural scenes are localized edge-like filters [2].

The projection coefficients for the linear combinations in the above methods can be either positive or negative, and such linear combinations generally involve complex cancellations between positive and negative numbers. Therefore, these representations lack the intuitive explanation from the relationship between parts and the whole.

Non-negative matrix factorization (NMF) [8] imposes the non-negativity constraints in learning basis images. The pixel values of resulting basis images, as well as coefficients for reconstruction, are all non-negative. By this way, only non-subtractive (or additive) combinations are allowed. This ensures that the components are combined to form a whole in an accumulative means. For this reason, NMF is considered as a procedure for learning a parts-based representation [8]. However, Li *et al*. [4] found that the non-negative basis components learned by NMF are not necessarily as localized as describe in the original NMF paper, at least for the ORL face database; moreover, the original NMF representation yields low recognition accuracy – lower than can be obtained by using the standard PCA method. Motivated by these observations, they proposed a local non-negative matrix factorization (LNMF) algorithm which optimizes the objective to learn truly localized, parts-based components. Their experimental results demonstrate that LNMF basis leads to much more stable recognition results when there are occlusions, better than the standard NMF and PCA methods.

In our method, LNMF is employed to learn parts-based components. We apply it on the input region (I) and (1-I) to get both bright local components and dark local components, suppose the input region (I) have the pixel value in the range of [0, 1]. Each local feature is calculated from a bright component and a dark one. We can then construct a face detector by selecting a small number of important features using AdaBoost from the overcomplete local feature set.

The rest of the paper is organized as follows. Section 2 gives a brief introduction of NMF and Constrained NMF. Section 3 describes how to get the local features used in our method. Constructing face classifier using AdaBoost is briefly introduced in Section 4. Experiments are presented in Section 5 followed by some conclusions.

2. NMF AND CONSTRAINED NMF

Given a set of N_T training images represented as an $n \times N_T$ matrix $\mathrm{X} = [x_{ij}]$, each column of which contains n non-negative pixel values. Denote a set of $m \le n$ basis images by an $n \times m$ matrix $\mathbf{W}$. Each image can be represented as a linear combination of the basis images (eigenvectors of unit length), and hence the (approximate) factorization

$$\mathrm{X} \approx \mathrm{W}\,\mathrm{H} \tag{1}$$

where H is the matrix of $m \times N_T$ coefficients or weights. Dimension reduction is achieved when $m < n$.

The PCA factorization requires that the basis images (columns of W be orthonormal and the rows of H be mutually orthogonal. It imposes no other constraints than the orthogonality, and hence allows the entries of W and H to be of arbitrary sign. The NMF and LNMF, however, allow only positive coefficients and thus additive combinations of basis components.

2.1 NMF

NMF imposes the non-negativity constraints instead of the orthogonality. As the result, the entries of *w* and *h* are all non-negative. This way, only additive combinations are allowed, and no subtractions can occur. This is believed to be compatible to the intuitive notion of combining parts to form a whole, and is how NMF learns a parts-based representation [8]. It is also consistent with the physiological fact that the firing rate is non-negative.

NMF uses the divergence of X from Y = W H , defined as

$$D(\mathbf{X} \| \mathbf{Y}) = \sum_{i,j} \left(x_{ij} \log \frac{x_{ij}}{y_{ij}} - x_{ij} + y_{ij} \right) \qquad (2)$$

as the measure of cost for factorizing X into WH. An NMF factorization is defined as a solution to the following constrained optimization problem

$$\min_{\mathbf{W},\mathbf{H}} D(\mathbf{X} \| \mathbf{WH}) \quad \text{s.t} \quad \mathrm{W}, \mathrm{H} \geq 0, \ \sum_i w_{ij} = 1 \ \ \forall j \qquad (3)$$

where W, H ≥ 0 means that all entries of W and H are non-negative.

2.2 Constrained NMF

The NMF model defined by (3) does not impose any constraints on the spatial locality. Therefore, minimizing the objective function can hardly yield a factorization which reveals local features in the data **X**. LNMF is aimed to improve the locality of the learned features by imposing additional constraints. Let $(\mathrm{W}^T\mathrm{W}) = \mathrm{U} = [u_{ij}]$, $(\mathrm{HH}^T) = \mathrm{V} = [v_{ij}]$. The following three additional constraints are imposed on the NMF basis:

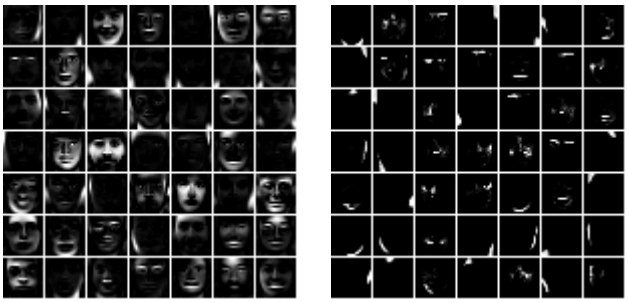
1. The number of basis components which is required to represent X should be minimized. This requires that a basis component should not be further decomposed into more components. Let w_j be a basis vector. Given the existing constraints $\Sigma_i w_{ij} = 1$ for all j, the value $\Sigma_i w_{ij}^2$ should be as small as possible so that w_j contains as many non-zero elements as possible. This constraint can be formulated as minimizing $\Sigma_i u_{ii}$.
2. To minimize redundancy between different bases, different bases should be as orthogonal as possible. This can be imposed by minimizing $\Sigma_{i \neq j} u_{ij}$.
3. Only basis containing most important information need to be retained. Given that every image in X is normalized into a range such as in [0, 1], the total "activity" on each component, i.e. the total squared projection coefficients summed over all training images, should be maximized. This is imposed by $\Sigma_i v_{ii} = \max$.

Incorporating the above constraints into the original NMF formulation, the new objective function for LNMF [4] is:

$$D(\mathbf{X} \| \mathbf{WH}) = \sum_{i,j} \left(x_{ij} \log \frac{x_{ij}}{y_{ij}} - x_{ij} + y_{ij} \right) + \alpha \sum_{i,j} u_{ij} - \beta \sum_i v_{ii} \qquad (4)$$

where $\alpha, \beta > 0$ are some constants. A LNMF factorization is defined as a solution to the problem (3) with (4).

A comparison shown in Figure 1 gives the different factorization results (image basis) of NMF and LNMF on our face database. LNMF basis are obviously more localized than NMF basis. One should note that because of the orthogonality constraint 2, the coefficient matrix H is no longer sparse in LNMF as it is in NMF. But this takes no effect on our approach since we only use the image basis.



(1) (2)

Figure 1 Factorization result of 49 basis on our face database by (1) NMF (Non-Negative Matrix Factorization) and (2) Constrained NMF. Obviously LNMF has more localized basis.

3. GETTING LOCAL FEATURES

Section 2 shows that LNMF can provide localized, parts-based representation of human faces. In this section, we will concentrate on using the representation to get efficient local features.

Investigating the Harr-like features used in Viola's [18] and Papageorgiou's [11] systems, we notice that differential operator is robust to varying lighting. Inspired by this, we desire to get local components that contain both bright and dark parts of the faces, and then put differential operator on bright and dark components to get the final value.

To achieve this, each sample (I) in **X** is mapped into **X**' as (1-I), suppose (I) have its pixel value in range [0, 1]. Then we apply LNMF on both the sample set **X** and **X**' to get two sets of basis, W and W', which could be used as bright and dark components, respectively.

This can be explained as below. Recall that in last section, the matrix V = (HH^T) indicates the energy relationship between the basis (include each basis itself). From the experiment we find that the values of the entries of **V** matrix are much closed to each other, implying that each basis contribute roughly the same to the whole data set. Thus we can not say individually which component is more "bright" than others. That is why we need to perform LNMF on the other sample set X'.

Given the two basis sets W and W', for each input region we can get two coefficient vector *h* and *h*'. The local feature set corresponding to the basis sets could be $\{h_i - h'_j\}$, $\forall i,j$. In practice, several local feature sets, correspond to different basis sets, are combined together to form a overcomplete feature set. In next section, AdaBoost is applied on the set to select important features and construct the classifier at the same time.

4. ADABOOST FOR FEATURE SELECTION

After the process described in previous section, we obtain an over-complete set of local features. Using the entire feature set is obviously infeasible in practice. Oppositely, we seek for an approach to select those most discriminating features. Viola [18] use a variant of AdaBoost to select features from an overcomeplete Harr-like feature set and train the classifier. The similar method is applied to our system.

The AdaBoost algorithm was first introduced in 1995 by Freund and Schapire [21]. In its original forms, the goal of AdaBoost is to improve the performance of any given classification algorithms via combining a collection of classification functions to form a stronger classifier. These classification functions, in the language of boosting, are usually called weak learners. The major idea of AdaBoost is to enforce the weak learners to focus on the examples misclassified by previous classifiers. It does this by adjusting the weight of each training sample. In the initial state, all weights are set equally but on each round of training, the weights of misclassified samples will be increased in the proportion of previous classification errors.

This greedy boosting procedure was adapted by Viola et al. to feature selection. The weak learner is restricted to a set of classification functions while each of which depends on only one single feature. For each feature, the weak learner determines an optimal threshold classification function, such that the number of misclassified examples is minimized.

The procedure of applying AdaBoost to feature selection [18] can be formulated as follows. Given a set of training examples $(x_1, y_1), (x_2, y_2), \ldots (x_m, y_m)$, where x_i represents 20×20 image patterns and $y_i = 0, 1$ for faces and nonfaces examples respectively, we assign a weight value w_i to each example (x_i, y_i) . Before the training all w_i are equal and the sums of all eights are normalized to unit. For each feature, we train a simple Bayesian classifier which is restricted to this single feature. The classification error is evaluated with respect to $w_i, \varepsilon_j = \sum w_i \left| h_j(x_i) - y_i \right|$. The classifier, h_t , with the lowest error ε_t is chosen as one component of the final strong classifier and its importance in final classification function is determined by classification rate. Subsequently all weights are updated in terms of the training error before next round of training.

Besides Viola's successful experience, the formal guarantees provided by the AdaBoost learning procedure are quite strong. Freund and Schapire [21] proved that the training error of the strong classifier approaches zero exponentially in the number of rounds. More importantly a number of results were later proved about generalization performance [22]. The key insight is that generalization performance is related to the margin of the examples, and that AdaBoost achieves large margins rapidly.

Using the same learning framework, we can now compare our learnt local features with Viola's Harr like features. We do this comparison on our training set which contains 5,000 face samples and 10,000 non-face samples. LNMF representations of dimensions 25, 36, 45, 47, 49, 51, 53, 55, 64, 81, 100 are computed from the training set to form a feature set with 37648 local features. From each features set, we select 200 features. The error ε_t of first 20 features is shown in Figure 2. Figure 3 shows the ROC curves of the two classifiers on our testing set which contains 2000 face samples and 5000 non-face samples.

5. EXPERIMENTAL RESULTS

This section describes the final face detection system, including training data preparation, training procedure, and the performance comparison with state-of-the-art face detection system.

5.1 Training Data Set

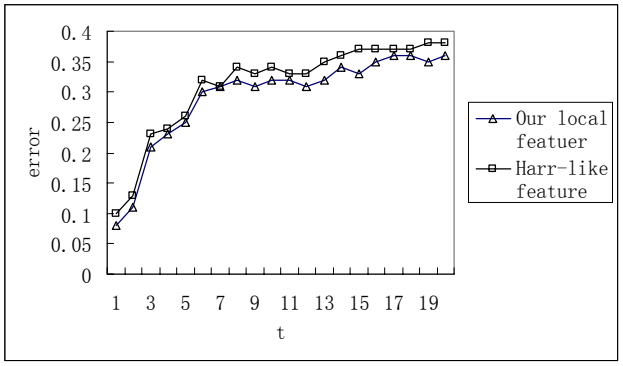


Figure 2. Comparison of our local feature set with Viola's Harr-like feature set using the first 20 features selected by AdaBoost.

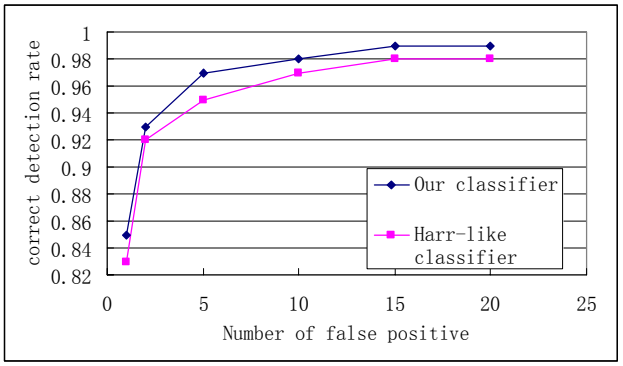


Figure 3. ROC curves of the two classifier using 200 features selected from our local feature set and Harr-like feature set

We collect the frontal face images from the database of CMU, Rockefeller, Umist, Corel and our own database. There are more than 7,000 faces in total. We select 5,000 of them as our positive training samples and 2,000 as testing samples. Each face image is resized into 20 × 20 and aligned by the center point of the two eyes and the horizontal distance between the two eyes.

For non-face training set, an initial 10,000 non-face samples were selected randomly from 15,000 large images which contain no face. The 5,000 testing non-face samples mentioned in section 4 are also randomly selected from the large images.

All samples, both in training set and in testing set, are processed by illumination compensation and histogram equalization to minimize the effect of different lighting conditions, as was done in Rowley's method.

5.2 Training phase

We use the similar feature selection framework with Viola 's method [18]. The final detector is a 29 layer cascade of classifier. We used 2 features in the first layer, 5 features in the second layer, and 20 features in three layers. In the fifteenth layers 200 features are used for training the classifier.

The initial 10,000 non-face samples are used to train the first three layers. In subsequent layers, non-face samples are obtained by scanning the partial cascade across large non-face images and collecting false positive samples. Different sets of nonface sub-windows are used in training the different classifiers to ensure that they are somewhat independent and use different features.

5.3 Testing phase

The face detector is tested on the images collected from the MIT+CMU test set [23]. For an input image, we scan each 20 × 20 sub-window exhaustively in both spatial and scale space, as was done is Rowley's system [14]. The starting scale is 1, the scale step is 1.25 and the spatial step is 1 pixel at each scale level. Results from different scale levels or spatial locations are merged to get the final result.

Figure 4. An example image of Output by our face detector

6. CONCLUSIONS

In this paper, we introduce a face detection approach by learning representative local features. The main difference from other face detection methods is that the local features are learnt from the training set instead of arbitrarily defined.

The learning procedure consists of two steps. First, Constrained NMF (Non-negative Matrix Factorization) is applied on the training set to get both bright and dark basis (components), then a overcomplete set of local features are construct upon that. This set of local features show better performance than Viola's exhaustively Harr-like feature set. Second, a learning algorithm based on

AdaBoost is used to select a small number of feature groups and yields extremely efficient classifiers. Experimental results show that the face detection performance on our test set is comparable to Viola's systems.

REFERENCES

[1] M.S. Barlett; H.M. Ladesand T.J. Sejnowski, "Independent component representations for face recognition", *Proc. SPIE* 3299, 528-539, 1998.

[2] A.J. Bell and T.J. Sejnowski, "The 'independent components' of natural scenes are edge filters", *Vision Research,* vol.37, pp.3327-3338, 1997.

[3] M.C. Burl and P. Perona, "recognition of planar object classes." pp.223-230, CVPR'96.

[4] S.Z. Li, X.W. Hou and H.J. Zhang, "Learning Spatially Localized, Parts-Based Representation", *IEEE CVPR*, 2001.

[5] A.J. Colmenarez and T.S. Huang, "Face detection with information-based maximum discrimination", *IEEE CVPR,* 782-787, 1997.

[6] P. Comon, "Independent component analysis – a new concept?", *Signal Processing*, vol.36, pp.287-314, 1994.

[7] C. Jutten and J. Herault, "Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture", *Signal Processing,* vol.24, pp.1-10, 1991.

[8] D. Lee; H.S. Seung, "Learning the parts of objects by non-negative matrix factorization", *Nature* V.401, 21, 788-791, Oct. 1999.

[9] N.K. Logothetis; D.L. Sheinberg, "Visual object recognition", *Annu. Rev. Neurosci.* 19, 577-621, 1996.

[10] S.E. Palmer, "Hierachical structure in perceptual representation", *Cogn. Psychol*. 9, 441-474, 1977.

[11] C.P. Papageorgiou; M. Oren and T. Poggio, "A general framework for object detection", ICCV '98.

[12] P. Penev and J. Atick, "Local feature analysis: A general statistical theory for object representation", *Neural Systems,* vol.7, no.3, 477-500, 1996.

[13] A. Pentland, B.Moghaddam, T.Starner. "View-Baased and Modular Eigenspaces for Face Recognition." CVPR 1994.

[14] H. Rowley; S. Baluja; and T. Kanade, "Neural Network-based face detection", *IEEE PAMI.,* 20(1), 23-38, 1998.

[15] H. Schneiderman and T. Kanade, "Probabilistic modeling of local appearance and spatial relationships for object recognition", *IEEE CVPR,* 45-51, 1998.

[16] H. Schneiderman, "A statistical approach to 3D object detection applied to faces and cars", Ph.D. thesis, CMU-RI-TR-00-06, May, 2000.

[17] K.-K. Sung and T. Poggio, "Example-based learning for view-based human face detection", *IEEE PAMI.,* 20(1), 39-51, 1998.

[18] P. Viola and M.J. Jones, "Robust real-time object detection", *Technical Report Series, Compaq Cambridge Research Laboratory, CRL* 2001/01, Feb. 2001.

[19] E. Wachsmuth; M.W. Oram; D.I. Perrett, "Recognition of objects and their component parts: responses of single units in the temporal cortex of the macaque", *Cereb. Cortex* 4, 509-522, 1994.

[20] M.-H. Yang; D. Roth, and N. Ahuja, "A SNoW-based face detector", *Advances in NIPS* 12, 855-861, 2000.

[21] Y. Freund and R.E. Schapire. "A decision-theoretic generalization of noline learning and an application to boosting", *Computational learning theory: Eurocolt'95* pp23-37, 1995.

[22] R.E. Schapire, Y. Freund, P. Bartlett, and W.S. Lee. "Boosting the margin: A new explanation for the effectiveness of voting methods", *Proc. Fourteenth International Conference on Machine Learning,* 1997.

[23] http://www.vasc.ri.cmu.edu/idb/html/face/frontal_images/index.html, MIT+CMU frontal face database.