Person X Detector

National Institute of Informatics at TERCVID 2004

Lizuo Jin, Shin'ichi Satoh, Fuminori Yamagishi, Duy Dinh Le, Masao Sakauchi
National Institute of Informatics, Tokyo 101-8430, Japan

Abstract This paper describes our participation in the NIST TERCVID 2004 retrieval evaluation. In the first-year effort for the TERCVID project, we only tackle Person X detection of the high-level feature extraction task. We design an automatic Person X detector using frontal faces in videos solely. We illustrate the architecture of Person X detector and the evaluation results in this paper.

1 Introduction

Automatic person identification in videos by machines such as Person X detection in broadcast news videos is still a great challenge, which deeply depends on the development of the research on computer vision, pattern recognition and machine learning. Generally we can build a machine system to differentiate persons in broadcast news videos using multimodal information such as video, audio, and text, etc., however, in the first-year participation in the TRECVID retrieval evaluation, we design an automatic Person X detector only utilizing frontal faces extracted from video sequences.

As far as we know, actually the state-of-art face recognition techniques [3, 23] are still far from the target to build a practical machine system for robust face recognition as powerful as human beings. We compare the up-to-date techniques on face detection, feature extraction and face recognition in the literature, and then enhance and integrate them into computer algorithms for identifying only several predetermined persons rather than build a general purpose face recognition or authentication system through retrieving large scale face databases.

In analogy with most face recognition systems, our Person X detector consists of 5 components (Fig. 1): image preprocessing, face detection, face alignment, feature extraction and face recognition. Hereafter, we elaborate each component of Person X detector in section 2. The evaluation results on TRECVID 2004 test set are given in section 3. In section 4, we conclude our current work on Person X detector and clarify the difficulties to be tackled in the future.

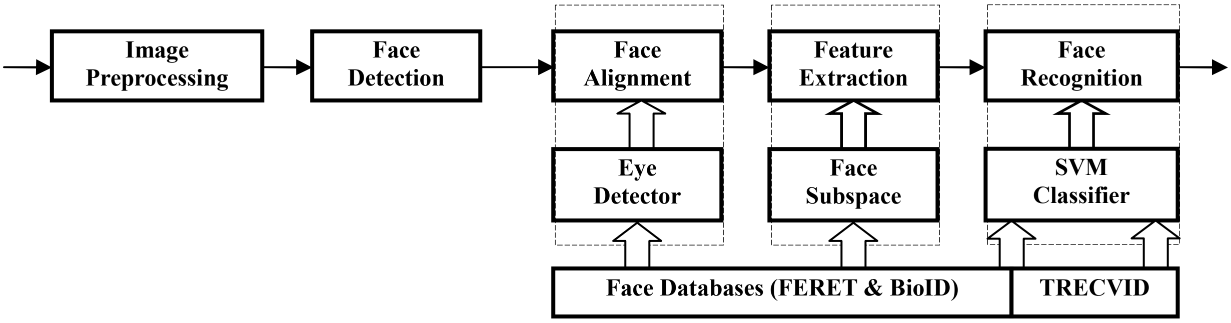


Figure 1 Architecture of Person X detector

2 Person X Detector

2.1 Image Preprocessing

Appearance based face detection [1, 2] and recognition methods [3, 23], such as the Eigenfaces method, the Fisherfaces method, neural network based methods, and support vector machine based methods, depend on pixel intensities deeply. When applied to images acquired under unconstrained conditions such as news video frames, they are usually unable to locate faces accurately due to the drastic variation of pixel intensities in face regions. Image enhancement by intensity transformation is very useful in alleviating such problem. Histogram equalization is currently one of the most popular techniques for face image enhancement. However, if applied to images with faces appearing on very light or very dark background, it may produce over dark or over light face regions, which may cause face detection failure; therefore image enhancement should adapt to pixel intensity distribution.

In paper [6], we present Entropy Error Rate (EER), an information-theoretic measure, to quantitatively estimate the information distribution of image pixels. This simple statistic depicts the asymmetry of information distribution of the darker pixels and the lighter pixels in the image.

$$EER = \frac{\overline{H}_D - \overline{H}_B}{S}, \qquad (1)$$

where

$$S = 4\left(\frac{I_{\max} - I_{mean}}{I_{\max} - I_{\min}}\right)\left(\frac{I_{mean} - I_{\min}}{I_{\max} - I_{\min}}\right), \tag{2}$$

$$\overline{H}_D = \frac{H_D}{I_{mean} - I_{\min} + 1}, \tag{3}$$

$$\overline{H}_B = \frac{H_B}{I_{\max} - I_{mean} + 1}. \tag{4}$$

S, called singularity, measures the relative position of the mean among the intensity range, which can describe the asymmetry of intensity distribution of the darker pixels and the lighter pixels in the image. H_D is the entropy of darker pixels whose intensities are below I_{mean}; H_B is the entropy of lighter pixels whose intensities are above I_{mean}; $\overline{H}_D$ and $\overline{H}_B$ are the corresponding average entropy of H_D and H_B.

Through analyzing the ridges among intensity histogram of the image, we can obtain some useful statistics on pixel intensity distribution and the valid range where pixel intensities representing relatively large objects may lie. Pixel intensities within the valid range are extended to the full range by linear stretching to make them vary in a more wide dynamic range and others are compressed to the least or highest gray level, which basically does not cause much information loss.

Furthermore, after chosen the format of intensity transform function, a power function, the image enhancement problem is explicitly converted into an optimization problem which searches for the optimal transform to minimize the EER of the intensified image. The transform using a power function is called gamma correction in image processing literature, so our method can be regarded as the optimal gamma correction as well.

To determine the effects of different enhancement techniques on face detection, we take the face detection approach proposed by Viola Jones et al. in [4] as the benchmark routine to conduct comparison experiments using both the Yale face database B [5] and our own movie face dataset which are captured under controlled and uncontrolled illumination conditions respectively. The experimental results show that after image enhancement preprocessing the accuracy of face detection improves significantly and our adaptive enhancement algorithm performs much better than histogram equalization and linear stretching.

2.2 Face Detection

The reliability of face detection strongly influences the performance and usability of Person X detector. Given a single video frame, an ideal face detector should be able to identify and locate all faces regardless of their positions, scales, orientations, illumination conditions, and expressions, etc.

Face detection can be performed based on multiple cues such as skin color, motion, facial shape, and facial appearance or a combination of them. Appearance and machine learning based methods are currently the most effective ones for face detection, such as the Eigenfaces method [15], neural network based methods [19], boosting based methods [4, 7], support vector machine based methods [15, 25], and Bayesian inference based methods [18, 20], etc [24]. The boosting based face detector proposed by Viola Jones et al. is currently one of the most fast and accurate frontal face detectors; since we only deal with frontal faces presently, we integrate it into our Person X detector for fast face detection. Actually we integrate the modified version of Viola Jones et al.'s algorithm which is embedded in Intel OpenCV b3.1 SDK [32]. For its sensitivity to illumination change, we take the above image enhancement preprocessing to improve performance of face detection when dealing with practical broadcast news video frames.

2.3 Face Alignment

Generally, the outputs of face detection are not geometrically normalized; if they are directly fed to later stages of face recognition system, it will inevitably lead to many recognition errors. Accurate extraction of facial features for efficiently representing faces in videos is crucial for Person X detector. The task of face alignment is to locate facial features such as eyes, nose, mouth, and face outline, etc. as precisely as possible, and further to normalize facial shape and texture.

There are several models and related methods for face alignment in the recent literature such as active shape model [11], active appearance model [9, 10, 26], direct appearance model [13] and 3D morphable model [8]. Though they are promising and advanced techniques for accurate face modeling and syntheses, they are still facing many difficulties such as the localization errors, sensitivity to initialization and illumination changes, and high computational cost, etc. We are not sure whether they can be effectively applied to automatically normalize faces in low-resolution videos. For geometrically normalizing faces accurately, it is important to locate two eyes precisely; therefore we design an automatic eye detector using machine learning techniques rather than apply above models and related alignment algorithms.

Due to the symmetry of human faces, we only design one detector for the left eye. To detect another eye, we only need to mirror the image first and then apply the detector again. The coordinate of the right eye should be tuned with respect to the width of the input face image.

The eye detector is based on a two-class support vector machine, and the kernel is the Gaussian RBF function.

The training set contains two parts: the positive set consists of 5000 left eye images which are cropped from face images in the FERET [17] and BioID [27] face database; the negative set consists of 5000 none eye images which are cropped from one natural image. The size of all the images in the training set is scaled to 21x17, and the pixel intensities are standardized to zero mean and unity variance. To implement the eye detector, we choose the LIBSVM 2.6 [28] to train the SVM classifier, and the optimal parameters for the classifier are searched by the Grid Search algorithm contained in this SVM development kit.

When applied to detect eyes practically, face images are first scaled to the predetermined size, and then shrunk to one out of 1.25 of the former one for several times. After each scaling, every region of size 21x17 is cropped by scanning the image pixel by pixel, and then standardized to zero mean and unity variance. All these regions are fed to the SVM classifier for eye/none eye classification. An eye may be detected several times at close locations or multiple scales. The number of multiple detections at a close location can be used as an effective indication for the existence of an eye at that location and to eliminate some false detection. A successful detection is confirmed if the number of multiple detections beyond a predetermined threshold and all the multiple detections are merged to a consistent one, which is located at the average position of them. When the positions of two eyes are detected, the candidate face images are rotated, scaled, translated and cropped to make the two eyes located at the predetermined pixels.

2.4 Feature Extraction

The goal of feature extraction is to create a low-dimensional representation of faces with good discriminatory power for classification. Currently there are dozens of methods [14, 29, 30] for feature extraction such as principle component analysis (PCA), linear discriminant analysis (LDA), independent component analysis (ICA), non-negative matrix factorization (NMF), local linear projection (LLP, a derivative of local linear embedding (LLE)), kernel principle component analysis (KPCA), kernel discriminant analysis (KDA), kernel independent component analysis (KICA), and Gabor wavelet transform (GWT), etc. Among them, PCA is one of the most wildly used tools for dimensionality reduction and feature extraction.

We conduct a comparison experiment of face recognition on the FERET database with PCA, ICA and PCA based whitening transform (PCAW) separately for feature extraction and a nearest neighbor classifier for pattern classification. Through simple mathematic derivation, we can find that Euclidean distance between any two feature vectors in PCAW subspace equals that in ICA subspace when ICA is implemented with the Fixed Point ICA algorithm [21] because Euclidean distance is invariant under an orthonormal transform. Through the experiment, we find that for face recognition PCAW performs as well as ICA if ICA is implemented with linear ICA algorithms [21, 22] and both of them outperform PCA when using Euclidean distance between feature vectors to measure the dissimilarity of objects. In fact PCAW is commonly used as a data preprocessing step for ICA, however, the orthonormal transform of ICA for high-order decorrelation to estimate global independent components contributes little to the improvement of pattern classification accuracy when using Euclidean distance as dissimilarity measure, so we only choose PCAW to build the low-dimensional subspace for face representation.

The average face recognition accuracy on four datasets fafb (1195 face images with expression change), fafc (194 face images with illumination change), pd1 (722 face images with aging) and pd2 (234 face images, a subset of pd1) of FERET database with PCA, ICA, PCAW and a nearest neighbor classifier are shown in figure 2. Three kinds of measures: L1 distance (City Block distance), L2 distance (Euclidean distance) and cosine (a derivative of Euclidean distance actually) are chosen to compute the dissimilarity or similarity between feature vectors in low-dimensional subspace. From the curves shown in figure 2, it is obvious that the optimal performance is obtained when using cosine and PCAW or cosine and ICA, and if using cosine or L2 distance for classification, the accuracy when using ICA is the same as that when using PCAW. Actually when feature vectors are normalized to unity norm, the recognition accuracy with cosine is equivalent to that with L2 distance. Also from the plots, we find the performance reaches the optimal when the dimensionality is reduced to 200. So we build a 200-dimensional PCAW subspace for face representation.

To build the PCAW subspace for face representation, we collect 3816 images from the FERET database and 1521 images from the BioID database, which are all frontal face images. These face images are rotated in plane, scaled to 52x60 and translated to make the two eyes locate at the predetermined pixels, and then all clutters are removed by a standard face shape mask. Furthermore, they are enhanced by histogram equalization and standardized to zero mean and unity variance to reduce the influence of illumination change. All the images are turned into vectors by scanning them row by row and then apply PCA analysis to calculate the mean vector, the first 200 eigenvalues and the corresponding eigenvectors. The PCAW subspace is spanned by the 200 vectors which are the product of the eigenvectors and the inverse square root of their corresponding eigenvalues.

Suppose the mean vector is $\overline{X}$, the eigenvalues are $\lambda_1,...,\lambda_{200}$, and the corresponding eigenvectors are $V_1,...,V_{200}$, then the base vectors $U_1,...,U_{200}$ which span the PCAW subspace are

$$U_i = \frac{V_i}{\sqrt{\lambda_i}}, i = 1,...200 . \quad (5)$$

They compose the transform matrix $\Phi = [U_1 \quad ... \quad U_{200}]$ of PCAW. For an unseen face observation X, its representation Y in PCAW subspace is obtained by

$$Y = \Phi^T (X - \overline{X}) . \qquad (6)$$

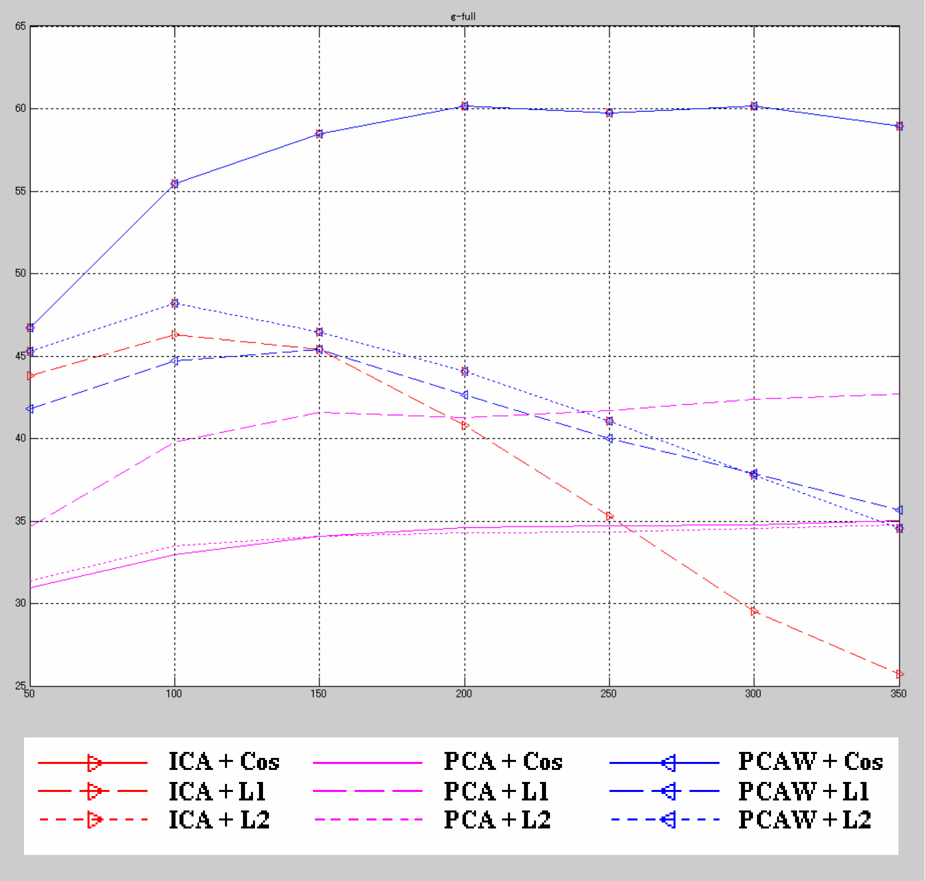


Figure 2 the average face recognition accuracy on FERET dataset with PCA, ICA and PCAW

2.5 Face Recognition

After obtaining the feature representation of faces, we need to train a classifier to learn a complex decision function to implement classification. As far as we know, optimizing the feature representation for the best discrimination could help reduce the complexity of decision function, and further a good classifier should be able to learn the separability between patterns. For example, for face recognition, the feature representation by LDA has better discriminatory power than that by PCA, and kernel machines basically outperform nearest neighbor classifiers when trained with the same feature representation and limited training samples.

Practically faces in videos are subjected to pose change, illumination change, and expression change, etc, and they are usually not linearly separable when projected to a linear low-dimensional subspace by a linear transform such as PCA, ICA, LDA, etc., we need more advanced tools to build the classifier. Meanwhile we face the small sample size problem when build a practical pattern classifier. We choose the kernel machine techniques developed in recent one decade to build the classifier, because it is promising to deal with the above difficulties.

Currently there are two types of architecture for designing a multi-class SVM: one-to-one and one-to-the-rest. For training Person X classifier, it is difficult to choose the one-to-one structure because over many individual classifiers will make it impossible to implement, so we select the other. Also the one-to-the-rest structure is possible for us to choose a general negative training set although it is maybe not the most effective option.

The Person X classifier is based on a two-class support vector machine and the kernel is the Gaussian RBF function. The training set contains two parts: the positive set for Bill Clinton consists of 1060 face images obtained by sampling 11 video segments of Clinton in the TRECVID dataset every 5 frames, each video segment lasts 20 seconds averagely; the positive set for Madeline Albright consists of 290 face images obtained by sampling from 4 video segments of Albright in the TRECVID dataset with the same rate, each video segment lasts 15 seconds averagely; the negative set consists of 3816 face images of other 1210 individuals in the FERET face database. The size of all the images is scaled to 52x60, all clutters are removed by a standard face shape mask, and the pixel intensities are enhanced by histogram equalization and standardized to zero mean and unity variance. All the standardized images are then transferred into vectors by scanning them row by row and projected to PCAW subspace to reduce the dimensionality. To geometrically normalize faces of Person X, we spent several days to manually label the eye positions in the images of Person X. Figure 3 shows the normalized faces of Clinton and Albright extracted from the same video segments respectively; the first row shows 10 samples of Bill Clinton, the second row shows 10 samples of Madeline Albright. Pose and expression change are obvious in these samples and the eye positions basically are moved to the same locations. Eye localization errors make the sample images of Albright vary much dramatically.

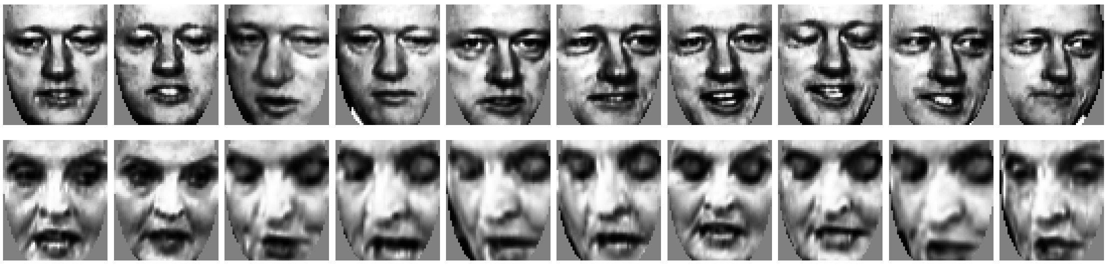


Figure 3 Normalized training samples of Person X

To implement the Person X classifier, we choose the algorithm SVM light 6.01 [31] to train the SVM classifier, and also the optimal parameters for the classifier are provided by the Grid Search algorithm contained in the LIBSVM 2.6 development kit.

Suppose the labeled training dataset of N observations of Person X is $\{Y_i, L_i\}, i = 1,...,N$, where $L_i \in \{-1,1\}$, the decision function of the SVM classifier can be described by

$$f(Y) = \mathrm{sgn}\{\sum_{i=1}^{N} L_i \alpha_i K(Y, Y_i) + b\}, \tag{7}$$

where $K(Y, Y_i) = \exp(-\frac{\|Y - Y_i\|^2}{2\sigma^2})$ is the Gaussian radial basis function.

To improve the accuracy of Person X classification, we change the decision function in (7) to

$$f(Y) = \begin{cases} +1, & \sum_{i=1}^{N} L_i \alpha_i K(Y, Y_i) + b > \delta_+ \\ -1, & \sum_{i=1}^{N} L_i \alpha_i K(Y, Y_i) + b < \delta_- \\ 0, & otherwise \end{cases}, \tag{8}$$

where $\delta_+ \geq 0$ and $\delta_- \leq 0$ are two predetermined thresholds. When $f(Y) = 0$, the classification result is obtained by temporary inference through a simple memorizing and forgetting algorithm.

Suppose the initial belief on Person X in one frame is B_0, the exciting threshold is T_e, the memorizing coefficient is C_m and the forgetting coefficient is C_f, the belief B_t on Person X in frame $t > 0$ is

$$B_t = \begin{cases} B_{t-1}(1 + C_m), & f(Y) = +1 \\ B_{t-1}(1 - C_f), & f(Y) = 0 \\ 0 & f(Y) = -1 \end{cases}. \tag{9}$$

When $B_t < T_e$, if $f(Y) \neq 1$ let $B_t = 0$ else let $B_t = B_0$. The belief F_t on the frame containing a frontal face when $M > 0$ faces are detected is

$$F_t = \begin{cases} B_t, & B_t \geq 1 \\ 1, & otherwise \end{cases}. \tag{10}$$

If $M = 0$ let $F_t = 0$. Then, the belief B_{px} on one video sequence of length $L > 0$ containing Person X is

$$B_{px} = \frac{\sum_{t=1}^{L} B_t}{\sum_{t=1}^{L} F_t}. \tag{11}$$

We conduct a belief computational comparison experiment on whether one shot containing Person X; figure 4 shows the variation of the belief values on whether it containing Bill Clinton. This shot is a part of video file 19981002_ABCa.mpg in TRECVID 2004 test set, which consists of 407 frames and whose inner code number is 29. We sample this shot every 30 frames and detect Bill Clinton from them. The processed frames are shown in figure 5; the computed belief values are given in table 2; the parameters setting for belief computation is list in table 1.

In figure 4, the circle represents the number of frontal faces detected; the pentagon represents $f(Y)$, which describes whether one face is discriminated as Bill Clinton by Person X classifier; the upward triangle represents the belief B_t on whether Bill Clinton appearing in one frame; the downward triangle represents the belief F_t on whether one frame containing frontal faces; the line represents the belief on whether one shot containing Bill Clinton; the dash line represents the frequency on Bill Clinton's appearing. From figure 4, we can find the belief computed by the temporary inference through a simple memorizing and forgetting algorithm is more stable than that by voting through computing the frequency of Bill Clinton's appearing.

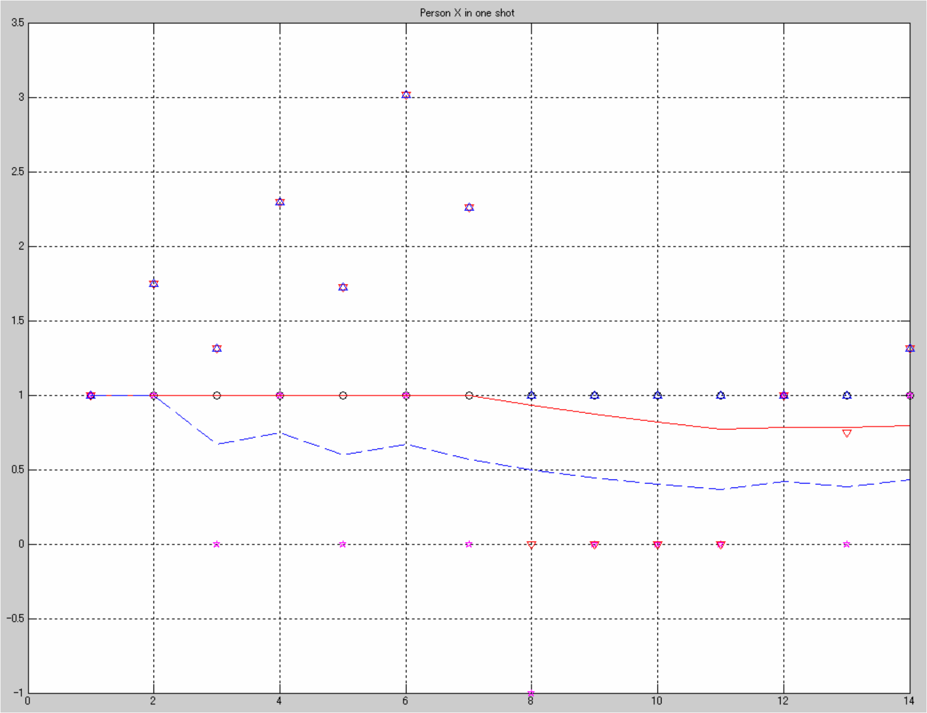


Figure 4 Variation of the belief values on whether one shot containing Bill Clinton

Table 1 Parameters setting for belief computation in figure 4

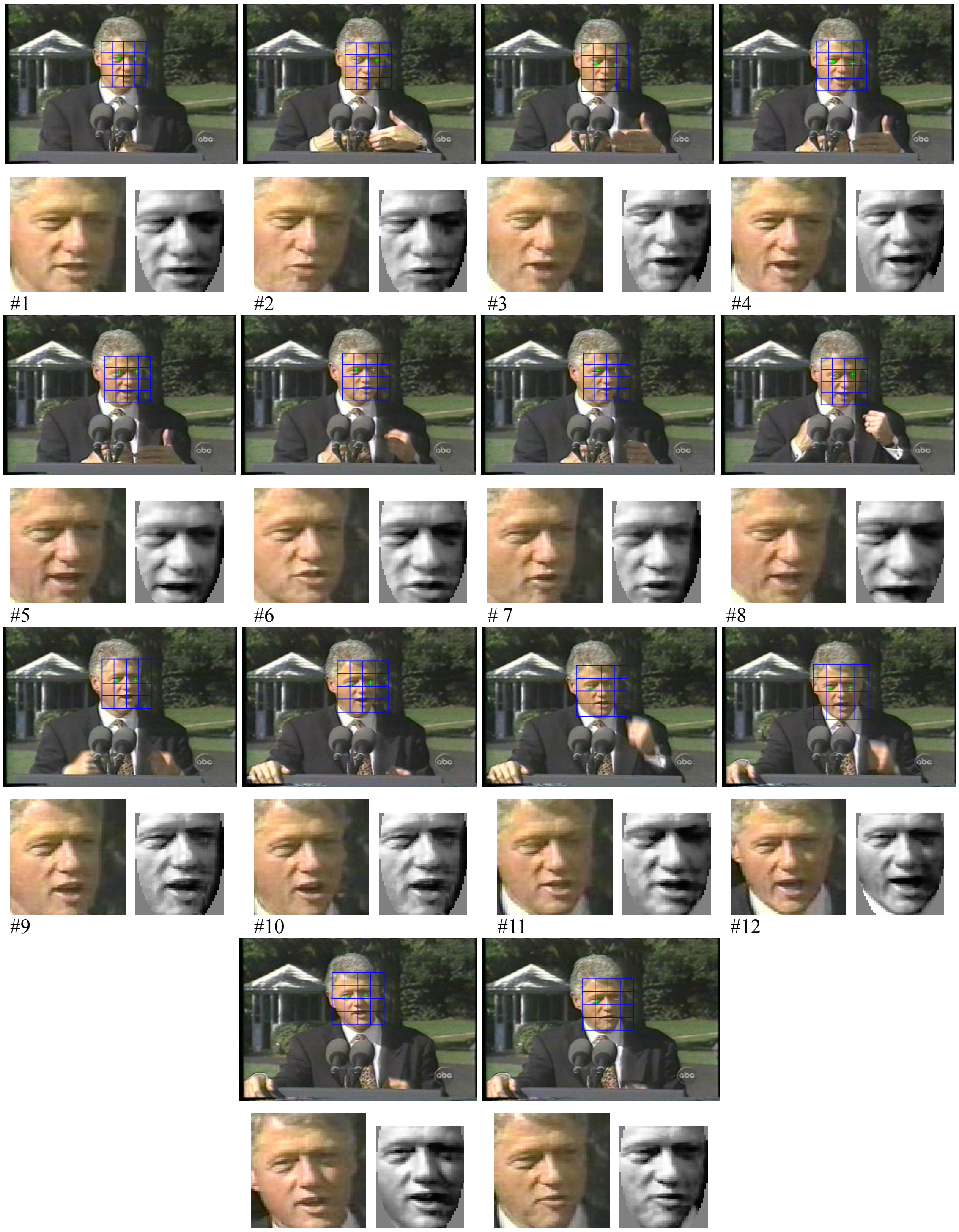
B_0	C_m	C_f	T_e	δ_+	δ_-	$Norm_{\min}$
1.00	0.75	0.25	0.50	0.10	-0.20	20.50

Due to the false positive of face detection and the localization error of eye detection, we need filter off the outliers before conducting Person X classification. In [15], a low-dimensional face representation is built based on PCA analysis, and further applied to frontal face detection. Similarly, suppose the reconstruction errors are small enough when face images are projected into PCAW subspace, the norm of the face feature vectors can be used as a probability measure of frontal faces. Based on this measure, we can remove outliers before executing Person X classification if they are larger than a predetermined threshold $Norm_{\min}$.

3 Evaluation results

To evaluate the performance of Person X detector, we conduct an experiment on identifying whether one shot containing Bill Clinton. This shot is the same as the one used for belief computational comparison experiment in figure 4. Figure 5 shows the results on detecting Bill Clinton in this shot; table 2 shows the detection and recognition results; the variation of belief values on Bill Clinton is shown in figure 4; the parameters setting for belief computation is given in table 1. In figure 5, the 1, 3, 5 and 7 rows show the original frames, in which blue rectangles label the detected face region and green crosses mark the detected eye positions; the 2, 4, 6 and 8 rows show detected face region images and the normalized face images.

From the experimental results in figure 5, it is obvious that some faces are not normalized correctly due to eye localization errors. Meanwhile, from table 2 we can find that as a result of face pose change and normalization error, it is unreliable to recognize Person X only based on one frame. Through temporary inference or voting, the recognition accuracy increases obviously; furthermore temporary inference through simple memorizing and forgetting algorithm is more robust than voting through computing the appearing frequency.



#1 #2 #3 #4 #5 #6 # 7 #8 #9 #10 #11 #12 #13 #14

Figure 5 Detection results of Bill Clinton within one shot on TRECVID 2004 test set

Table 2 Detection results of Bill Clinton within one shot on TRECVID 2004 test set

No.	**Faces**	$f(Y)$	B_t	F_t	B_{px}	**Frequency**
1	1	+1	1.00000000	1.00000000	1.00000000	1.00000000
2	1	+1	1.75000000	1.75000000	1.00000000	1.00000000

3	1	0	1.31250000	1.31250000	1.00000000	0.66666667
4	1	+1	2.29687500	2.29687500	1.00000000	0.75000000
5	1	0	1.72265625	1.72265625	1.00000000	0.60000000
6	1	+1	3.01464844	3.01464844	1.00000000	0.66666667
7	1	0	2.26098633	2.26098633	1.00000000	0.57142857
8	1	-1	0.00000000	1.00000000	0.93035080	0.50000000
9	1	0	0.00000000	1.00000000	0.86977188	0.44444444
10	1	0	0.00000000	1.00000000	0.81659975	0.40000000
11	1	0	0.00000000	1.00000000	0.76955427	0.36363636
12	1	+1	1.00000000	1.00000000	0.78210738	0.41666667
13	1	0	0.75000000	1.00000000	0.78044874	0.38461538
14	1	+1	1.31250000	1.31250000	0.79438965	0.42857143

We perform 10 experiments of Person X detection with different parameters setting on TRECVID 2004 test set which involves 409 shots containing Bill Clinton and 29 shots containing Madeline Albright; the results on Bill Clinton detection are shown in table 3, where $FN_{\min}$ means the least number of frames in one shot containing frontal faces detected, TSR means the number of total shots returned, NSR means the number of shots with features returned, AP means the average precision, and PSF means the precision at shots with feature.

Table 3 Detection results of shots containing Clinton on TRECVID 2004 test set

Run	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
$FN_{\min}$	4			3			5			1
B_0	1.00									
C_m	0.75	0.75	0.50	0.75	0.75	0.50	0.75	0.75	0.50	0.75
C_f	0.25	0.25	0.50	0.25	0.25	0.50	0.25	0.25	0.50	0.25
T_e	0.50	0.25	0.50	0.50	0.25	0.50	0.50	0.25	0.50	0.50
TSR	1059	1052	1048	1192	1184	1180	945	938	935	1466
NSR	116	115	115	127	126	126	104	103	103	140
AP	0.1213	0.1196	0.1255	0.1263	0.1247	0.1303	0.1125	0.1106	0.1165	0.0923
PSF	0.2445	0.2421	0.2494	0.2567	0.2518	0.2543	0.2274	0.2249	0.2274	0.2372
Recall	**Precision**									
0.0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
0.1	0.5600	0.5600	0.6000	0.5301	0.5301	0.5726	0.5714	0.5714	0.6200	0.2602
0.2	0.3860	0.3814	0.3773	0.4511	0.4481	0.4409	0.3402	0.3294	0.3682	0.2602
0.3	0.0000	0.0000	0.0000	0.1352	0.1341	0.1341	0.0000	0.0000	0.0000	0.1549
0.4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.5	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Shots	**Precision at n Shots**									
5	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000

10	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000	0.2000
20	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000	0.4000
100	0.5300	0.5300	0.5900	0.5100	0.5100	0.5500	0.5500	0.5500	0.6200	0.3500
200	0.4000	0.4000	0.3950	0.4200	0.4200	0.4200	0.3850	0.3800	0.3800	0.1900
500	0.2120	0.2100	0.2080	0.2160	0.2140	0.2200	0.1900	0.1880	0.1880	0.2180
1000	0.1150	0.1140	0.1140	0.1240	0.1240	0.1240	0.1040	0.1030	0.1030	0.1310

From table 3, we find that changing the exciting threshold T_e basically does not affect detection accuracy; better performance is obtained when the memorizing coefficient C_m equals the forgetting coefficient C_f, which means stronger memorizing ability could not yield higher detection accuracy. Since the Person X detection is based on face sequence analyzing, the number of faces involved in the belief computation also influence detection accuracy. The more is the number of faces, the higher is the precision of the first 100 ranks, but the lower is the average precision; however, if the least number of faces (only one) involved in belief computation, the detection precision becomes much lower. Currently the best average precision of our Person X detector for Bill Clinton is 13.03% and the best precision of the first 100 ranks is 62.00%, which is slightly higher than the median-level precision.

We also perform another experiment on Madeline Albright detection on TRECVID 2004 test set and the results are much lower than the median-level precision, the best average precision is only 0.13%. We analyze the results and conclude that it maybe due to over many outliers yielded by normalization errors in the positive training set.

4 Conclusions

In the first-year participation in the NIST TRECVID retrieval evaluation, we only tackle Person X detection of the high-level feature extraction task. We design an automatic Person X detector for identifying video segments containing Person X. To identify Person X, we apply linear subspace projection by PCAW for low-dimensional features extraction and a two-class SVM for pattern classification. To compute the belief whether one video segment containing Person X, we introduce a simple memorizing and forgetting algorithm for temporary inference.

From the evaluation results, we find Person X detection accuracy deeply depends on the quality of face normalization, and the performance of SVM classifier is also affected by the outliers contained in the positive training set. Meanwhile pose change of faces in real videos decreases the detection accuracy greatly. All these problems need to be tackled in the future version of Person X detector.

References

[1] M. -H. Yang, D. J. Kriegman, and N. Ahuja, “Detecting Faces in Images: A Survey”, IEEE Trans. on PAMI, vol. 24, no. 1, pp. 34–58, 2002.

[2] E. Hjelmas, and B. K. Low, “Face Detection: A Survey”, CVIU, vol. 83, pp. 236–274, 2001.

[3] W. Zhao, R. Chellappa, P. J. Philips, and A. Rosenfeld, “Face Recognition: A Literature Survey”, ACM Computing Surveys, vol. 35, no. 4, pp. 399–458, 2003.

[4] P. Viola and M. Jones, “Robust Real-time Object Detection”, Proc. of ICCV, vol. 20, no. 11, pp. 1254–1259, 2001.

[5] Website of Yale face database B - http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html.

[6] Lizuo Jin, Shin’ichi Satoh, and Masao Sakauchi, “A Novel Adaptive Image Enhancement Algorithm for Face Detection”, Proc. of ICPR, vol. 4, pp. 843–848, 2004.

[7] Stan Z. Li, ZhenQiu Zhang, “FloatBoost Learning and Statistical Face Detection”, IEEE Trans. on PAMI, vol. 26, no. 9, pp. 1112–1123, 2004.

[8] V. Blanz and T.Vetter, “A Morphable Model for the Synthesis of 3D Faces”, Proc. of SIGGRAPH, vol. 1, pp. 187–194, 1999.

[9] T. Cootes and C. Taylor, ”Constrained active appearance models”, Proc. of ICCV, vol. 1, pp. 748–754, 2001.

[10] T. F. Cootes, G. J. Edwards, and C. J. Taylor, “Active appearance models”, Proc. of ECCV, vol. 2, pp. 484–498, 1998.

[11] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, “Active shape models: Their training and application”, CVGIP: Image Understanding, vol. 61, pp. 38–59, 1995.

[12] T. F. Cootes, K. N. Walker, and C. J. Taylor, “View-based active appearance models”, Proc. of ICFGR, pp. 227–232, 2000.

[13] X. W. Hou, S. Z. Li, and H. J. Zhang, “Direct appearance models”, Proc. of CVPR, vol. 1, pp. 828–833, 2001.

[14] B. Moghaddam. “Principal manifolds and probabilistic subspaces for visual recognition”, IEEE Trans. on PAMI, vol.24, no. 6, pp. 780–788, 2002.

[15] B. Moghaddam and A. Pentland, “Probabilistic visual learning for object representation”, IEEE Trans. on PAMI, vol. 7, no. 6, pp. 696–710, 1997.

[16] E. Osuna, R. Freund, and F. Girosi, “Training support vector machines: An application to face detection”, Proc.

of CVPR, vol. 1, pp. 130–136, 1997.

[17] Website of FERET face database - http://www.itl.nist.gov/iad/humanid/feret/feret_master.html.

[18] D. Roth, M. Yang, and N. Ahuja, "A snow-based face detector", Proc. of NIPS, 2000.

[19] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection", IEEE Trans. on PAMI, vol. 20, no. 1, pp. 23–28, 1998.

[20] Henry Schneiderman and Takeo Kanade, "Object Detection Using the Statistics of Parts", IJCV, vol. 56, no. 3, pp.151-177, 2004.

[21] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis", IEEE Trans. on Neural Networks, vol. 10, no. 3, pp. 626–634, 1999.

[22] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," Neural Comput., vol. 7, no. 6, pp. 1129–1159, 1995.

[23] Seong G. Kong, Jingu Heo, Besma R. Abidi, Joonki Paik, and Mongi A. Abidi, "Recent advances in visual and infrared face recognition - a review", CVIU, vol. 95, no. 3, 2004.

[24] Duy Dinh Le and Shin'ichi Satoh, "Fusion of local and global features for efficient object detection", Applications of Neural Networks and Machine Learning in Image Processing IX, IS&T/SPIE Symposium on Electronic Imaging, 2005.

[25] Yongmin Li, Shaogang Gong, Jamie Sherrah, and Heather Liddell, "Support vector machine based multi-view face detection and recognition", Image and Vision Computing, vol.22, pp. 413–427, 2004.

[26] Iain Matthews and Simon Baker, "Active Appearance Models Revisited", IJCV, vol. 60, no. 2, pp. 135-164, 2004.

[27] Website of BioID face Database - http://www.humanscan.de/support/downloads/facedb.php.

[28] Website of LIBSVM 2.6 - http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html.

[29] M. Bartlett, J. Movellan, and T. Sejnowski, "Face recognition by independent component analysis", IEEE Trans. on Neural Network, vol. 13, no. 6, pp. 1450–1464, 2002.

[30] J. Draper, K. Baek, M. Bartlett, and J. Beveridge, "Recognizing faces with PCA and ICA", CVIU, vol. 91, pp. 115–137, 2003.

[31] Website of SVM Light – http://svmlight.joachims.org.

[32] Website of Intel OpenCV SDK – http://www.intel.com/research/mrl/research/opencv.