

“Pattern Recognition”

Shervin Halat

98131018

Project

(Final)

Project Topic

**An Efficient Joint Formulation for
Bayesian Face Verification**

‘Summary of Article’

Introduction:

This article mainly focuses on a sub-problem of face recognition, called “face verification” which aims to decide if two given faces belong to mutual identity or not. This problem is different from “face identification” in which the goal is to specify the identity of a given face from a set of trained identities. This article introduces enhancements to the previous versions of Bayesian face verification methods which is discussed step by step in next parts of this report.

Brief flashback to previous Bayesian face verification algorithm:

In the previous versions, the face verification problem amounts to classifying the difference $\Delta = x_1 - x_2$ (where x_1 and x_2 denote two faces to be verified) as intra-personal variation or extra-personal variation using the log-likelihood ratio statistic:

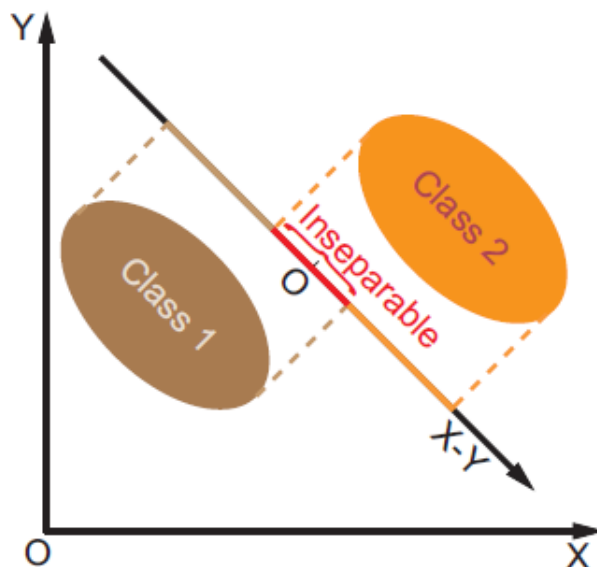
(H_I represents the intra personal hypothesis that two faces x_1 and x_2 belong to the same subject, and H_E is the extra-personal hypothesis that two faces are from different subjects.)

$$r(x_1, x_2) = \log \frac{P(\Delta | H_I)}{P(\Delta | H_E)}$$

Why new algorithm introduced:

Considering differences of two objects, for example in 2D, can be modelled as projection of points in 2D onto a 1D space then applying classification. This is actually what is done in previous algorithms. While such a projection may retain some important discriminative information, it may nonetheless reduce the separability of the classes. Therefore, the power of Bayesian face framework may be limited by discarding the discriminative information. This the main reason why the new method was followed in this article. The mentioned problem is illustrated in the graph below:

(The 2-D data is projected to 1-D by $x-y$. The two classes, which are separable in the original joint representation, become inseparable after the projection.)



Approach proposed by this article:

To tackle the issues mentioned in previous part, it is proposed to directly model the full joint distribution of $\{x_1, x_2\}$ for the face

verification problem in a similar Bayesian framework. This approach begins with introducing an appropriate parametric prior on face representations where each face feature is expressed as the summation of two independent Gaussian variables, one related to identity and another that captures intra-personal variations. In this parametric model, parameters are estimated using the EM algorithm, while low-rank and invariance properties helps with efficient training and testing phases. The briefly discussed approach is described more detailed in the steps below:

Step 1:

The same as previous models it is assumed that appearance of a face can be well-approximated by two additive factors: identity and intra-personal variation which, as mentioned before, are two independent gaussian variables, μ and ε respectively.

$$x = \mu + \varepsilon$$

where x is the observed face feature vector with the mean of all faces subtracted, μ represents identity component, and ε is remaining intra-personal variations. variables μ and ε are distributed independently as $N(0, S_\mu)$ and $N(0, S_\varepsilon)$ respectively, where S_μ and S_ε are two unknown covariance matrixes. Together these distributions constitute a prior distribution on faces.

Under hypothesis H_I , $P(x_1, x_2/H_I)$ is a zero-mean Gaussian with covariance of:

$$\Sigma_I = \begin{bmatrix} S_\mu + S_\varepsilon & S_\mu \\ S_\mu & S_\mu + S_\varepsilon \end{bmatrix}$$

In contrast, assuming H_E , the covariance matrix of the distribution $P(x_1, x_2 | H_E)$ is:

$$\Sigma_E = \begin{bmatrix} S_\mu + S_\varepsilon & 0 \\ 0 & S_\mu + S_\varepsilon \end{bmatrix}$$

after a series of linear algebra manipulations, the log-likelihood ratio $r(x_1, x_2)$ can be obtained by following equation:

$$r(x_1, x_2) = \log \frac{P(x_1, x_2 | H_I)}{P(x_1, x_2 | H_E)} = x_1^T A x_1 + x_2^T A x_2 - 2x_1^T G x_2$$

where

$$A = (S_\mu + S_\varepsilon)^{-1} - (F + G)$$

and F and G satisfies:

$$\begin{bmatrix} F + G & G \\ G & F + G \end{bmatrix} = \begin{bmatrix} S_\mu + S_\varepsilon & S_\mu \\ S_\mu & S_\mu + S_\varepsilon \end{bmatrix}^{-1}$$

Now that the general parameterized equation of log-likelihood ratio is obtained its time for next step, that is estimation of the unknown parameters of S_μ and S_ε using EM algorithm.

Step 2:

Now the unknown parameters that we need to learn are the covariance matrices of identity and intra-person variation

$\Theta = \{ S_{\mu}, S_{\varepsilon} \}$. EM algorithm is applied to estimate the covariances by maximizing the loglikelihood function. That is:

$$\min_{S_{\mu}, S_{\varepsilon}} - \sum_i \log P(\mathbf{x}_i | S_{\mu}, S_{\varepsilon}) \quad \text{s.t.} \quad S_{\mu} \succeq 0, S_{\varepsilon} \succeq 0$$

Then based on generative process and drawing assumed ' m_i ' samples from normally distributed covariates $\{ S_{\mu}, S_{\varepsilon} \}$, and stack all samples together in a vector (\mathbf{h}) we have:

$$\mathbf{x}_i = [\mu_i + \varepsilon_{i1}; \dots; \mu_i + \varepsilon_{im}]$$

Therefore, for the identity and intra-person variations stacked into a column vector (\mathbf{h}) we have:

$$\mathbf{h}_i = [\mu_i; \varepsilon_{i1}; \cdot \cdot \cdot; \varepsilon_{im}]$$

with covariance matrix of:

$$\Sigma_{\mathbf{h}_i} = \text{diag}(S_{\mu}, S_{\varepsilon}, \dots, S_{\varepsilon})$$

Hence, from generative process the likelihood function of subject 'i' is:

$$P(\mathbf{x}_i | S_{\mu}, S_{\varepsilon}) = N(0, \Sigma_{\mathbf{x}_i}), \quad \text{where } \Sigma_{\mathbf{x}_i} = Q_i \Sigma_{\mathbf{h}_i} Q_i^T$$

Due to non-convexity of the problem, an EM algorithm is described for the purpose of minimizing

$$P(\mathbf{x}_i | S_{\mu}, S_{\varepsilon})$$

Ignoring complicated computations, the final results of E-step and M-step of the EM algorithm is as follows:

E-step: (Calculation of expectation of the hidden variables of each subject)

$$\begin{aligned}
 F &= S_{\varepsilon}^{-1} \\
 G &= -(m_i S_{\mu} + S_{\varepsilon})^{-1} S_{\mu} S_{\varepsilon}^{-1} \\
 E[\mu_i] &= S_{\mu} (F + m_i G) \sum_j x_{ij} \\
 E[\varepsilon_{ij}] &= S_{\varepsilon} F x_{ij} + S_{\varepsilon} G \sum_j x_{ij} \\
 E[\mu_i \mu_i^T] &= E[\mu_i] E[\mu_i]^T \\
 E[\varepsilon_{ij} \varepsilon_{ij}^T] &= E[\varepsilon_{ij}] E[\varepsilon_{ij}]^T
 \end{aligned}$$

M-step: (Model parameters update)

$$\begin{aligned}
 S_{\mu} &= \frac{1}{n} \sum_i E[\mu_i \mu_i^T] \\
 S_{\varepsilon} &= \frac{1}{k} \sum_{i,j} E[\varepsilon_{ij} \varepsilon_{ij}^T]
 \end{aligned}$$

Where ‘n’ is the number of subjects and ‘k’ is the total number of all images of all subjects.

$$k = \sum_i m_i$$

Note:

Some considerations for more efficient training and testing phases of the algorithm are discussed in the article (considering the high ‘computational complexity’ of mentioned equations), which will be discussed in ‘My Work’ section of the report.

“My Work”

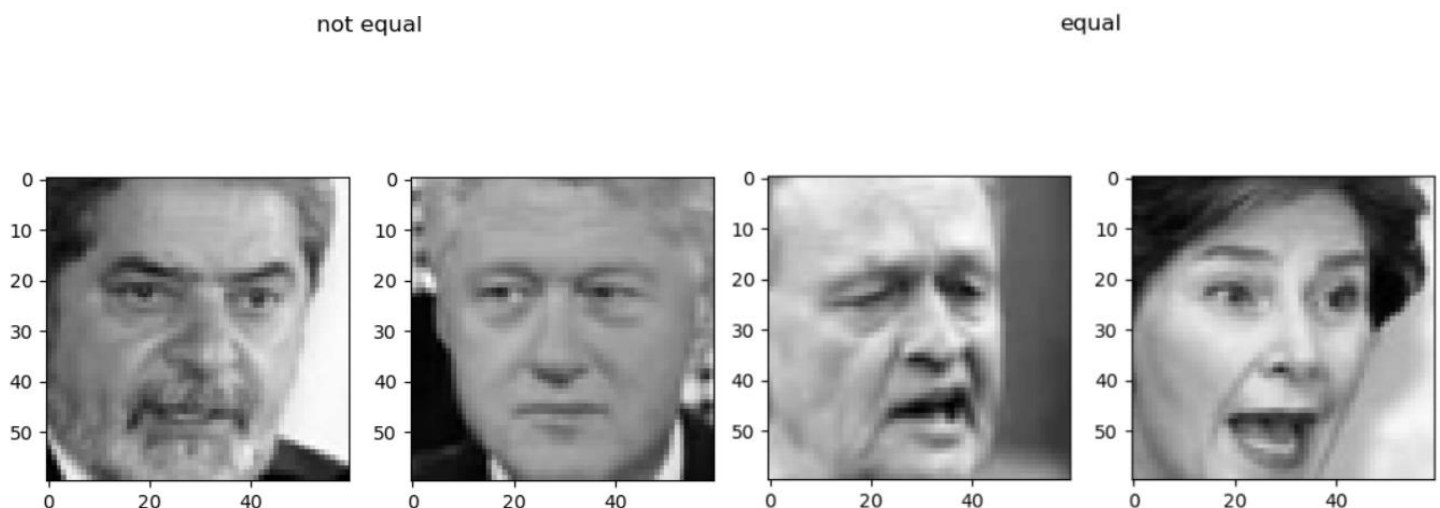
In this section, thoroughly considering steps discussed in both ‘theory’ and ‘experiments’ parts of the article, the implementation of the article in python language will be explained.

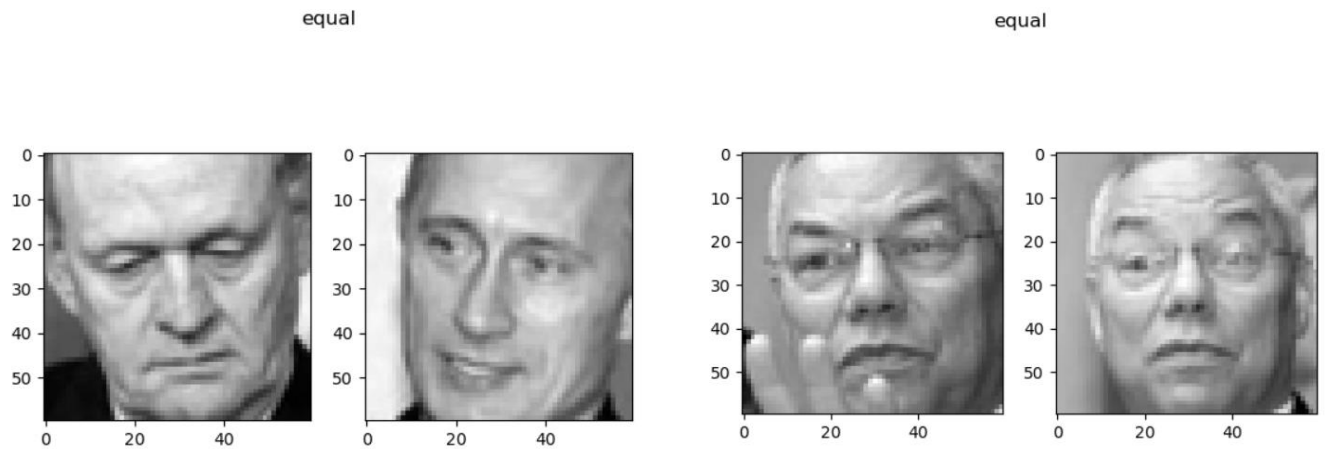
(Related script is “SPRproject.py”)

According to the experiment section of the article the training data (face images) were preprocessed which will be discussed later.

For implementation, the free dataset, known as LFW, was downloaded which contains multiple face images. Approximately, 140 images were used for training phase in my work which is considerably much less than hundreds of thousands training images which were implemented in article’s experiments. This is due to the runtime of the program! Also it should be noted that the dimensions of face images were reduced to 60x60 in my work because of the runtime.

The accuracy obtained in my work was approximately 60% for multiple runs by 8 iterations for EM algorithm and dimensionality reduction to 600 with PCA. Some sort of results:



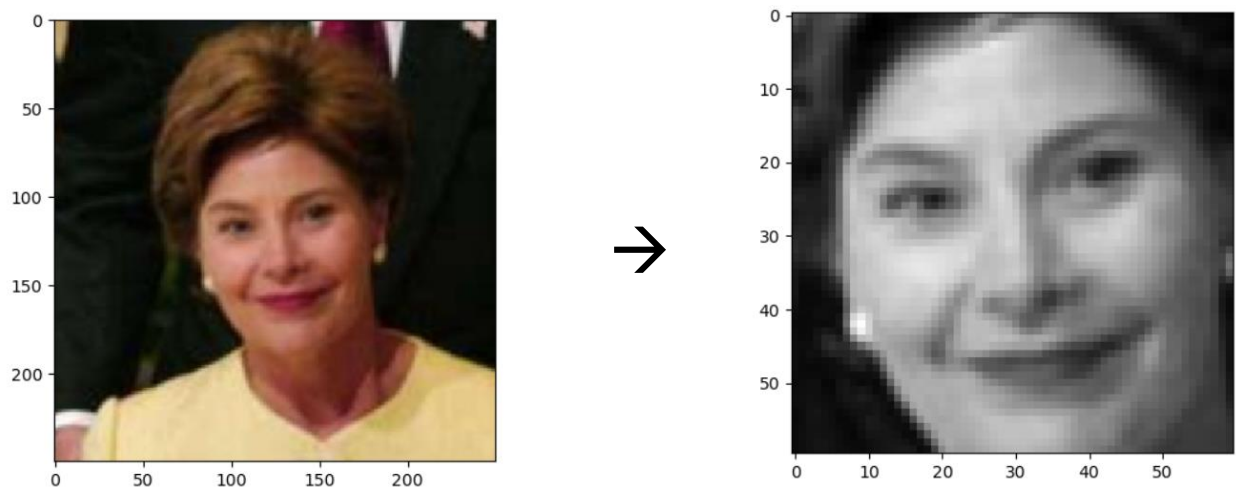


The steps taken in Script are as follows:

Step 1:

“Face Extraction”

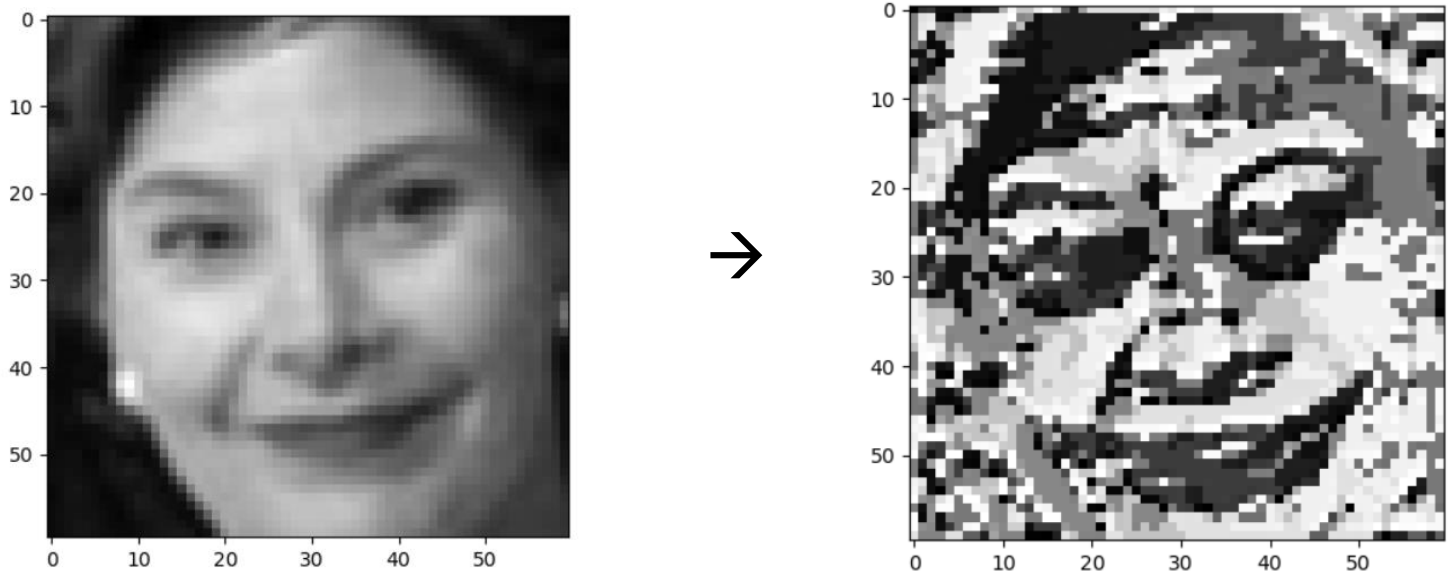
Before applying LBP feature of each face image, the face of each image was detected and cropped from the main image and then resized to 60x60 dimensions in order to have equal image dimensions. An example is illustrated below:



Step 2:

“LBP”

Then, LBP feature was applied to each image then all images were concatenated to form feature vectors.



Step 3:

“PCA”

Now, in order to reduce dimensionality of feature vectors, PCA was applied to these set of vectors and the dimensions were reduced to 400 by projection on 400 eigenvectors corresponding to 400 largest eigenvalues.

Now pre-processing phase is finished and we jump into training phase

“Training Phase”

Step 4:

The main goal in Training Phase is to estimate values of parameters S_μ and S_ε . For this purpose, initial values should be assigned to these covariance matrixes in order to initiate EM algorithm. According to the context, following equations may be used to compute acceptable initial values **which were implemented in the script**.

$$S_\varepsilon = \frac{1}{mn} \sum_i \sum_j (x_{ij} - \bar{x}_i)(x_{ij} - \bar{x}_i)^T$$
$$S_\mu = \frac{1}{n} \sum_i \bar{x}_i \bar{x}_i^T,$$

Step 5:

Now with obtained initial values, 6 iterations (as it is said to be enough for convergence according to the article) of EM algorithm are executed for the convergence of S_μ and S_ε . The EM algorithm equations are as follows:

Calculate the expectation of the hidden variables of each subject (E step):

$$\begin{aligned} F &= S_\varepsilon^{-1} \\ G &= -(m_i S_\mu + S_\varepsilon)^{-1} S_\mu S_\varepsilon^{-1} \\ E[\mu_i] &= S_\mu (F + m_i G) \sum_j x_{ij} \\ E[\varepsilon_{ij}] &= S_\varepsilon F x_{ij} + S_\varepsilon G \sum_j x_{ij} \\ E[\mu_i \mu_i^T] &= E[\mu_i] E[\mu_i]^T \\ E[\varepsilon_{ij} \varepsilon_{ij}^T] &= E[\varepsilon_{ij}] E[\varepsilon_{ij}]^T \end{aligned}$$

Update the model parameter (M step):

$$\begin{aligned} S_\mu &= \frac{1}{n} \sum_i E[\mu_i \mu_i^T] \\ S_\varepsilon &= \frac{1}{k} \sum_{i,j} E[\varepsilon_{ij} \varepsilon_{ij}^T] \end{aligned}$$

“Test Phase”

Step 6:

After obtaining S_μ and S_ε covariances, its time for final step, that is “Test Phase”. For this purpose, initially, the A and G parameters of log-likelihood equation (which is discussed in previous section) is calculated then the similarity (or log-likelihood equation) equation will be computed by two given sample vectors. Lastly, the positive output of similarity equation denotes equal identities of samples and negative output denotes non-equal identities.

$$A = (S_\mu + S_\varepsilon)^{-1} - [(S_\mu + S_\varepsilon) - S_\mu(S_\mu + S_\varepsilon)^{-1}S_\mu]^{-1}$$

$$G = -(2S_\mu + S_\varepsilon)^{-1}S_\mu S_\varepsilon^{-1}$$

$$r(x_1, x_2) = 2g_1^T g_2 - a_1^T a_1 - a_2^T a_2$$