# Jensen-Shannon Boosting Learning for Object Recognition

Xiangsheng Huang

CASIA-SAIT HCI Joint Lab, Institute
of Automation, Chinese Academy of Science
Beijing, China 100080

Stan Z. Li

Institute of Automation
Chinese Academy of Science
Beijing, China 100080

Yangsheng Wang

CASIA-SAIT HCI Joint Lab,Institute
of Automation, Chinese Academy of Science
Beijing, China 100080

## Abstract

In this paper, we propose a novel learning method, called Jensen-Shannon Boosting (JSBoost) and demonstrate its application to object recognition. JSBoost incorporates Jensen-Shannon (JS) divergence [2] into AdaBoost learning. JS divergence is advantageous in that it provides more appropriate measure of dissimilarity between two classes and it is numerically more stable than other measures such as Kullback-Leibler (KL) divergence (see [2]). The best features are iteratively learned by maximizing the projected JS divergence, based on which best weak classifiers are derived. The weak classifiers are combined into a strong one by minimizing the recognition error.

JSBoost learning is demonstrated with face object recognition using a local binary pattern (LBP) [13] based representation. JSBoost selects the best LBP features from thousands of candidate features and constructs a strong classifier based on the selected features. JSBoost empirically produces better face recognition results than other AdaBoost variants such as RealBoost [12], GentleBoost [5] and KLBoost [7], as demonstrated by experiments.

## 1 Introduction

Learning for feature selection and classification has become a focus of research in pattern recognition. Many algorithms have been developed for learning an effective and efficient classifier. For example, Linear Discriminant Analysis (LDA) is powerful tool for data reduction and feature extraction. However, LDA assumes that the two classes are both Gaussian distributed, and maximizes the distance between two class over the variation within each class. It does not work well for non-Gaussian problems.

Recently, AdaBoost learning algorithms have been proposed as an effective means for solving the above problem [4, 12]. AdaBoost not only selects the most discriminative and efficient features and constructs a strong classifier, it is also capable of dealing with nonconvex classification problems. It is shown that AdaBoost algorithms minimize an exponential function of the margin over the training set [4, 12]. Recent studies have proven that AdaBoost is statistically optimal in terms of posterior or likelihood [5], thus explaining AdaBoost from statistical viewpoints. However, learning weak classifiers remains a task for practice.

In this paper, we propose a novel learning method, called Jensen-Shannon Boosting (JSBoost) and demonstrate its application to object recognition. A major objective in this paper is to propose an AdaBoost learning procedure in which effective and stable weak classifiers are learned based on an appropriate discriminant measure. In this regard, JSBoost incorporates Jensen-Shannon (JS) divergence [2] into AdaBoost learning. The most discriminating features are sought by maximizing the JS divergence.

JS divergence is advantageous in that it is numerically more stable than other measures such as Kullback-Leibler (KL) divergence [2]. Also, it is well defined to provide appropriate measure of differences between two classes due to its proved upper bound [6]. In contrast, KL divergence, as used in [7], is undefined when the histogram of either class contains empty bins. This problem is avoided in JS divergence [6]. Moreover, no general upper bound exists for KL divergence in term of variational distance [14, 3].

JSBoost learning is demonstrated with face object recognition using a local binary pattern (LBP) [13] based representation. LBP features are invariant with respect to uniform and are more stable for illumination changes [13]. While thousands of candidate features can be produced for a small image, JSBoost selects the best LBP features from a large pool and constructs a strong classifier based on the selected features. During the learning, the intra-personal and extra-personal differences are used to convert multi-class problem into one of two classes [9], while JSBoost algorithm is meant for two class problems. Experiments are performed using FERET and BANCA face databases. JSBoost is empirically produces better face recognition results than other AdaBoost variants such as RealBoost [4], GentleBoost [5] and KLBoost [7], as demonstrated by experiments.

The rest of this paper is organized as follows. We introduce the main theory of JSBoost in Section 2. In Section 3, JSBoost-Based face Object Recognition is presented. Experimental results are shown in Section 4. We conclude the paper in Section 5.

# 2 JSBoost Learning

Provided that we are given a set of labelled examples $\{x_i, y_i\}_{i=1}^N$ where $x_i \in R^d$ and $y_i \in \{-1, +1\}$, and are asked to give a decision $y$ for any input feature $x \in R^d$. Rather than using the raw data, we may extract some features by mapping the raw data to some feature space using some mapping functions $\phi : R^d \rightarrow R$, where $\phi()$ can be linear (e.g. wavelet transforms) or nonlinear (e.g. LBP filters). After a best feature $\phi_i$ is learned ($i = 1, \dots, k$), it is mapped by a discriminant function $\varphi_i() : R \rightarrow R$. For each feature $\phi_i$, we may obtain from ensembles of the two classes histograms as estimated distributions for the positive (intra-personal) and negative (extra-personal) examples, $h_i^+(\phi_i(x))$ and $h_i^-(\phi_i(x))$. One would like the discriminant function $\varphi_i$ to be such that $\varphi_i(\phi_i(x)) > 0$ if $h_i^+(z) > h_i^-(z)$, or $\varphi_i(\phi_i(x)) < 0$ otherwise. An appropriate choice is the logistic function

$$\varphi_i() = \frac{1}{2} \log \frac{h_i^+()}{h_i^-()} \tag{1}$$

In [12], it is proved that such a function maximize the margin or the posterior.

Our goal is to learn the features $\phi_i$ reliably using good discriminant function $\varphi_i$ so as to construct a classifier that produces the lowest error rate with a minimum number of features. The classification function thus becomes

$$F(x) = \mathit{sign}[\sum_{i=1}^{k} \frac{1}{2} \log \frac{h_i^+(\phi_i(x))}{h_i^-(\phi_i(x))}] \tag{2}$$

with $sign() \in \{-1, 1\}$ being the class indicator function.

## 2.1 JS Feature Selection

There are two things to learn in the classification function: the set of best features $\{\phi_i\}$ and combining coefficients $\{\alpha_i\}$. Liu and Shum [7] adopted the maximizing information gain criterion to maximize the Kullback-Leibler (KL) divergence of the positive and negative histograms projected on the feature. KL is computed as follows

$$KL(\phi_i) = \int h_i^+(\phi_i(x)) \log \frac{h_i^+(\phi_i(x))}{h_i^-(\phi_i(x))} d\phi_i(x) \tag{3}$$

where $h_i^+(\phi_i(x))$ and $h_i^-(\phi_i(x))$ are the histograms of the positive and negative examples with weights $W_i(x_i^+)$ and $W_i(x_i^-)$, respectively. It is well known that $KL(\phi_i)$ is non-negative, additive but not symmetric. To obtain symmetry, Liu [7] used a symmetric version (SKL divergence) as follows

$$\begin{aligned} SKL(\phi_i) = \textstyle\int \{[h_i^+(\phi_i(x)) - h_i^-(\phi_i(x))]\cdot \\ \log \tfrac{h_i^+(\phi_i(x))}{h_i^-(\phi_i(x))}\} d\phi_i(x) \end{aligned} \tag{4}$$

However, it should be noted that KL divergence is undefined if $h_i^+(\phi_i(x)) = 0$ or $h_i^-(\phi_i(x)) = 0$.

JS divergence [2], defined below, overcome this problem:

$$\begin{aligned} JS(\phi_i) = \textstyle\int \{h_i^+(\phi_i(x))\cdot \\ \log \tfrac{h_i^+(\phi_i(x))}{\frac{1}{2}[h_i^+(\phi_i(x)) + h_i^-(\phi_i(x))]}\} d\phi_i(x) \end{aligned} \tag{5}$$

While $JS(\phi_i)$ is not a symmetric measure, a symmetric version can be defined as

$$\begin{aligned} SJS(\phi_i) = \textstyle\int \{h_i^+(\phi_i(x)) \log \tfrac{h_i^+(\phi_i(x))}{\frac{1}{2}[h_i^+(\phi_i(x)) + h_i^-(\phi_i(x))]} \\ + h_i^-(\phi_i(x)) \log \tfrac{h_i^-(\phi_i(x))}{\frac{1}{2}[h_i^+(\phi_i(x)) + h_i^-(\phi_i(x))]}\} d\phi_i(x) \end{aligned} \tag{6}$$

JS divergence as a measure turns out to have numerous desirable properties. From the Shannon inequality, we know that $JS(\phi_i) \geq 0$ and $JS(\phi_i) = 0$ $if\ and\ only\ if$ $h_i^+(\phi_i(x)) = h_i^-(\phi_i(x))$, which is essential for a measure of difference. Another important property of $JS(\phi_i)$ is that the variational distance serves as an upper bound [6]. For binned histograms it is computed as

$$\begin{aligned} D_{SJS} = \textstyle\sum_{l=0}^{L-1} \{h_i^+(l) \log \tfrac{h_i^+(l)}{\frac{1}{2}[h_i^+(l) + h_i^-(l)]} \\ + h_i^-(l) \log \tfrac{h_i^-(l)}{\frac{1}{2}[h_i^+(l) + h_i^-(l)]}\} \end{aligned} \tag{7}$$

where $L$ is the number of bins. From the equality given in Equ.7, we have

$$\begin{aligned} D_{SJS} &= \textstyle\sum_{l=0}^{L-1} h_i^+(l) \log [h_i^+(l)] \\ &\quad - \textstyle\sum_{l=0}^{L-1} h_i^+(l) \log \tfrac{h_i^+(l) + h_i^-(l)}{2} \\ &\quad + \textstyle\sum_{l=0}^{L-1} h_i^-(l) \log [h_i^-(l)] \\ &\quad - \textstyle\sum_{l=0}^{L-1} h_i^-(l) \log \tfrac{h_i^+(l) + h_i^-(l)}{2} \\ &= 2 \cdot H(\tfrac{h_i^+ + h_i^-}{2}) - H(h_i^+) - H(h_i^-) \end{aligned} \tag{8}$$

where $H$ is the Shannon entropy function. Equ.8 provides one possible physical interpretation of JS-divergence with information theoretic viewpoint. Furthermore, JS-divergence is more numerically stable for empirical distributions [2]. So we adopt symmetric JS-divergence (SJS) as a measure. The most discriminating and efficient feature that maximizes the JS-divergence

$$\phi_k^* = \arg\max_{\phi_i} JS(\phi_i) \tag{9}$$

is called JS feature.

## 2.2 Boosting by Sample Re-weighting

By means of JS feature pursuit and parameter learning, we can achieve the maximum drop of the recognition error by adding a feature. But how can we continue to drive these two steps to gradually decrease the training error? The answer lies in selecting complementary features using boosting strategy. From the previous learned classifier, we could increase the weight of misclassified samples while reducing the weight of recognized samples, and then learn a best feature to discriminate them. Assume that at step $(k-1)$ the weights of the positive and negative samples are $W_{k-1}(x_i^+)$ and $W_{k-1}(x_i^-)$, respectively. We use the soft identify function $f(x)$ instead of the for updating the weights, similar to what has been used in GentleBoost, to avoid frequent switching of the weight of the samples around the border. At step $k$, the weights are adjusted according to the following:

$$\begin{array}{l} W_k(x_i^+) = \frac{1}{Z^+} W_{k-1}(x_i^+) \exp\{-\beta_k y_i^+ f_{k-1}(x_i^+)\} \\ W_k(x_i^-) = \frac{1}{Z^-} W_{k-1}(x_i^-) \exp\{-\beta_k y_i^- f_{k-1}(x_i^-)\} \end{array} \tag{10}$$

where $Z^+$ and $Z^-$ are normalization factors for $W_k(x_i^+)$ and $W_k(x_i^-)$, respectively. Note the sequence $\beta_k$ controls how fast to adopt the weight. $\beta_k$ is chosen as

$$\beta_k = \log \frac{1-\varepsilon_k}{\varepsilon_k} \tag{11}$$

where $\varepsilon_k$ is the training error of current classifier. Fig.1 shows the flowchart of JSBoost learning.

```
0. (Input)
    (1) Training examples $\{(x_1, y_1), \ldots, (x_N, y_N)\}$,
        where $N = a + b$; of which $a$ examples have $y_i = +1$
        and $b$ examples have $y_i = -1$;
1. (Initialization)
    $w_i = \frac{1}{2a}$ for those examples with $y_i = +1$ or
    $w_i = \frac{1}{2b}$ for those examples with $y_i = -1$.
2. (Forward Inclusion)
  For $k = 1, \ldots, K$
    (1) Select JS feature $\phi_k$ by Jensen-Shannon divergence
        with weights $w_i$ via Equ.7. Let $f_k(x) = \frac{1}{2} \log \frac{h_k^+(\phi_k(x))}{h_k^-(\phi_k(x))}$
    (2) Update weight $w_i \leftarrow w_i \cdot \exp(-\beta_k \cdot y_i \cdot f_k(x_i))$, $i = 1$,
        $\ldots, N$, and normalize weights so that $\sum_i w_i = 1$
3. (Output)
    $F(x) = \begin{cases} 1, & if \ \sum_{i=1}^k \frac{1}{2} \log \frac{h_i^+(\phi_i(x))}{h_i^-(\phi_i(x))} > 0 \\ -1, & otherwise \end{cases}$
```

Figure 1: Jensen-Shannon boosting learning Algorithm.

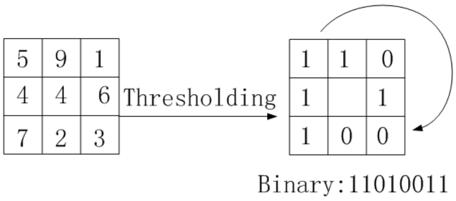

Figure 2: The basic LBP operator

# 3 JSBoost-Based Object Recognition

An image of object is represented by thousands of Local Binary Pattern (LBP) features. Multi-class problem is converted to one of two classes using the intra- and extra-class presentation [9]. JSBoost is applied to select most discriminative features and construct a strong classifier.

## 3.1 LPB Feature Representation

The original LBP operator, introduced by Ojala [10], is a powerful method of texture description. The operator labels the pixels of an image by thresholding the $3 \times 3$-neighborhood of each pixel with the center value and considering the result as a binary number. Later the operator was extended to use neighborhoods of different size [11]. Using circular neighborhoods and bilinearly interpolating the pixel values allow any radius and number of pixels in the neighborhood. Then the histogram of the labels can be used as a texture descriptor. An illustration of the basic LBP operator is shown in Fig.2.

An extension to the original operator is to use so called uniform patterns [11]. A Local Binary Pattern is called uniform if it contains at most two bitwise transitions from 0 to 1 or vice versa when the binary string is considered circular. Ojala noticed that in their experiments with texture images, uniform patterns account for a bit less than 90% of all patterns when using the (8,1) neighborhood and for 70% in (16,2) neighborhood. We use the following notation for the LBP operator: $LBP_{P,R}^{u2}$ means using the operator in a neighborhood of $P$ sampling points on a circle of radius $R$. Superscript $^{u2}$ stands for using uniform patterns and labelling all remaining patterns with a single label.

In this work, $LBP_{8,2}^{u2}$ is applied to extract LBP code for each pixel of face images, generating LBPfaces. All feature values are quantified into 59 bins according to uniform strategy. A histogram of the labelled image $f_l(x,y)$ can be defined as

$$H_i = \sum_{x,y} I\{f_l(x,y) = i\}, \qquad i = 0, \ldots, n-1 \tag{12}$$

in which $n$ is the number of different labels produced by the LBP (in this work, LBP coefficients are quantified into 59 bins, so $n$ is 59) and

$$I\{A\} = \begin{cases} 1, & A\ is\ true \\ 0, & A\ is\ false \end{cases} \quad (13)$$

The LBP histogram contains information about the distribution of the local micro-patterns, such as edges, spots and flat areas, over the whole image. For effective face representation, one should retain also spatial information. For this purpose, the face image is scanned with a scalable subwindow, thus a sequence of regions $R_0, R_1, \ldots, R_{m-1}$ is generated. The LBP histograms are extracted using LBP codes from these subwindows:

$$H_{i,j} = \sum_{x,y} I\{f_l(x,y) = i\} I\{(x,y) \in R_j\} \quad (14)$$

The histogram provides an effective description of the face on two different levels of locality: The labels for the histogram contain information about the patterns at the pixel-level while the labels summed over a small region provide information at the regional level [13].

### 3.2 Intra/Extra-Class Representation

The face recognition task is considered as a two-class problem by classifying every two face images as intra-personal or extra-personal ones and outputting a similarity describing how confident two face images are of one person [9]. The Chi square distance between corresponding LBP histograms [13] are used as discriminative features for this intra/extra-personal classification:

$$\chi^2(S,M) = \sum_{i}^{n} \frac{(S_i - M_i)^2}{(S_i + M_i)} \quad (15)$$

where $S$ and $M$ are two histograms, and $n$ is the bin number of histogram.

When the image has been divided into regions, it can be expected that some of the regions contain more useful information than others in terms of distinguishing between people. For examples, eyes seem to be an important cue in human face recognition. To take advantage of this, JSBoost is applied to select intra/extra feature and set them with different weight based on the importance of the information it contains.

### 3.3 Feature Selection and Classifier Learning

The set of LBP features for intra-personal and extra-personal classification, over 7000, contains much redundant information. In this paper, We propose to use JSBoost to select most significant LBP features from a large feature set, then construct weak classifiers each of which is based on one of the selected features, finally boost the weak classifiers into a stronger classifier.

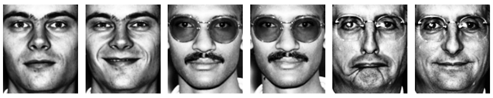

Figure 3: Some examples of preprocessed face images

The basic form of JSBoost is for two class problems. A set of $N$ labelled training examples is given as $(x_1, y_1), \ldots, (x_N, y_N)$, where $y_i \in \{+1, -1\}$ is the class label for the example $x_i \in \mathbb{R}^n$. In this work, $y_i = +1$ indicates that $x_i$ is an intra-personal example. Every training example is associated with a weight. JSBoost assumes that a procedure is available for learning a sequence of *weak classifiers* $f_k(x)$ ($k = 1, 2, \ldots, K$) from the training examples, with respect to the distributions $w_i^{(k)}$ of the examples. During the learning process, the weights are updated dynamically in such a way that more emphasis is placed on hard examples which are erroneously classified previously. The JSBoost algorithm is shown in Fig. 1. The JSBoost learning procedure is aimed to derive $f_k(x)$. A *stronger classifier* is a linear combination of the $K$ weak classifiers

$$F(x) = \sum_{k=1}^{K} f_k(x) \quad (16)$$

The magnitude $|F(x)|$ indicates the confidence.

## 4 Experiments

### 4.1 Experiments on The FERET Database

The proposed method was tested on the FERET $fa$ $fb$ face database, and the training set is also from the training set of the FERET database, which includes 1002 images of 429 subjects. All images are cropped to 142 pixels high by 120 pixels wide and rectified according to eye positions the FERET data provided. Histogram normalization and Gaussian smoothing are used to preprocess all cropped images. The cropped and preprocessed images are illustrated in Fig.3. $795$ intra-personal image pairs and $500,706$ extra-personal image pairs are generated using the training set.

By shifting and scaling the sub-window, 7,350 LBP histograms are extracted from each face image. And we get 7,350 candidate features for the intra-extra personal classification by computing the Chi square distance between corresponding LBP histograms of each image pairs. JSBoost was applied on the positive sample set of $795$ intra-personal image pairs and the negative sample set of $500,706$ extra-personal image pairs. The first four sub-windows learned, from which the LBP histograms are extracted, are shown in Fig.4.

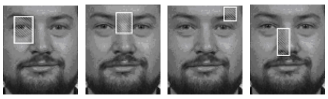

Figure 4: The first four sub-windows from which the Chi square distance between corresponding LBP histograms are obtained

To test the efficiency of our proposed method, several comparative experiments, namely RealBoost, GentleBoost, and KLBoost, were tested on the probe set $fb$ with the gallery $fa$ of the FERET database. The training and testing results are shown in Table 1.

Table 1 Experimental results based on Boosting (FN: Feature number, RR: Recognition rate)

| | RealBoost | GentleBoost | KLBoost | JSBoost |
|---|---|---|---|---|
| FN | 32 | 30 | 22 | 18 |
| RR | 97.9% | 98.0% | 98.1% | 98.4% |

It should be noted that Boosting (RealBoost, GentleBoost, KLBoost, and JSBoost) local binary pattern features obtains higher recognition rate (97.9%-98.4%) than weighted local binary pattern features (97%) [13]. Furthermore, JSBoost obtain higher recognition rate than other Boost, while using much less features. The rank curves of the final recognition results are plotted in Fig.5. It should be noted that the CSU implementations of the algorithms whose results we introduced here do not achieve the same figures as in original FERET test due to some modifications in the experimental setup. Our approach has achieved the upper bound recognition performance shown in Fig.5.

## 4.2 Experiments on The BANCA Database

The BANCA database contains 52 subjects (26 males and 26 females). Each subject participated to 12 recording sessions in different conditions and with different cameras. Session 1-4 contain data under controlled conditions while sessions 5-8 and 9-12 contain degraded and adverse scenarios respectively. In the BANCA protocol, 7 different distinct experimental configurations have been specified, namely Matched Controlled (MC), Matched Degraded (MD), Matched Adverse (MA), Unmatched Degraded (UD), Unmatched Adverse (UA), Pooled test (P) and Grand test (G) [1].

In this paper, we use the G protocol to test the propose algorithm. It should be pointed out that it is impossible for us to evaluate the proposed algorithm performance by comparing its result to the competition algorithms's results, such as Tsinghua's or Nottingham's results [8]. Because the competition models are trained using other databases as pointed out in [8], and we do not know which database are used to train their models.

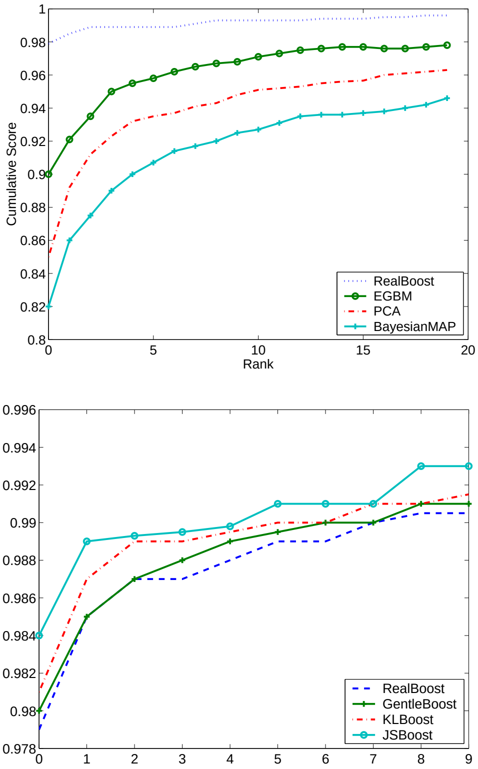

Figure 5: Rank curves for the $fb$ probe sets

To evaluate the performance of the novel approach, several comparative experiments, namely RealBoost, GentleBoost, and KLBoost, were tested on the BANCA database. In this experiment, 7,350 LBP histograms are obtained from each face image. And we get 7,350 candidate features for the intra-extra personal classification by computing the Chi square distance between corresponding LBP histograms of each image pairs. During training, 2730 intra examples and 73125 extra examples generated by 26 subject are used. We found that it is hard to convergence using RealBoost or GentleBoost in training stage.

To assess the algorithm performance, the Weighted Error Rate $(WER)$ for test data of groups $G1$ and $G2$ at the three different values of $R(0.1, 1, 10)$ [1] is calculated. $WER$ is defined as:

$$WER(R) = \frac{P_{FR} + R \cdot P_{FA}}{1 + R} \tag{17}$$

where $P_{FR}$ is the false recognition rate and $P_{FA}$ is the false acceptance rate. For each group and at each operating point, there are in total 1170 true client claims and 1560 imposter attacks. The experimental results are shown in Table 2. From the experimental results, we found that JSBoost obtain the lowest weighted error rate, while using the least features.

Table 2 Experimental results using KLBoost and JSBoost

(RB: RealBoost, GB:GentleBoost, KL:KLBoost, JS:JSBoost, FN:Feature number)

| | Train Set | FN | R=0.1(WER) | R=1 | R=10 |
|---|---|---|---|---|---|
| RB | G1 | 1050 | 11.42 | 17.56 | 8.43 |
| GB | G1 | 997 | 10.78 | 17.43 | 7.86 |
| KL | G1 | 107 | 8.89 | 11.98 | 3.38 |
| JS | G1 | 95 | 7.60 | 10.64 | 3.13 |
| RB | G2 | 1038 | 11.24 | 17.46 | 8.14 |
| GB | G2 | 985 | 10.53 | 17.31 | 7.67 |
| KL | G2 | 97 | 8.85 | 14.04 | 4.18 |
| JS | G2 | 85 | 6.67 | 10.74 | 3.80 |

## 5 Conclusion

In this work, we propose a novel learning algorithm based on Jensen-Shannon divergence called JSBoost. JS divergence is advantageous in that it provides more appropriate measure of dissimilarity between two classes and it is numerically more stable than other measures such as Kullback-Leibler (KL) divergence. JSBoost incorporates Jensen-Shannon (JS) divergence into AdaBoost learning. The best features are iteratively learned by maximizing the projected JS divergence, based on which best weak classifiers are derived.

JSBoost learning is demonstrated with face object recognition using a local binary pattern (LBP) [13] based representation. The multi-class problem of face recognition is transformed into a two-class one of intra-personal and extra-personal classification [9]. JSBoost selects the best LBP features from thousands of candidate features and constructs a strong classifier based on the selected features.

Experimental results on FERET $fa$ $fb$ and BANCA databases have proven that JSBoost seek more stable and generalization features for pattern recognition and produces better recognition rate than other AdaBoost variants such as RealBoost, GentleBoost and KLBoost.

## References

[1] E. Bailly-Bailliere, S. Bengio, F. Bimbot, M. Hamouz, J. Kittler, J. Mariethoz, J. Matas, K. Messer, V. Popovici, F. Poree, B. Ruiz, and J. P. Thiran. “The banca database and evaluation protocol”. In *The fouth International Conference on Audio- and Video-based Biometric Person Authentication, AVBPA’2003*, Berlin, Germany, June 2003.

[2] Y. Bubner, J. Puzicha, C. Tomasi, and J. M. Buhmann. “Empirical evaluation of dissimilarity measures for color and texture”. *Computer Vision and Image Understanding*, 84:25–43, 2001.

[3] D.Kazakos and T. Cotsidas. “A decision theory approach to the approximation of discrete probability densities”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(2):61–67, Jan 1980.

[4] Y. Freund and R.E. Schapire. “A decision-theoretic generalization of on-line learning and an application to boosting”. *Journal of Computer and System Sciences*, 55(1):119–139, August 1997.

[5] J. Friedman, T. Hastie, and R. Tibshirani. “Additive logistic regression: a statistical view of boosting”. *The Annals of Statistics*, 28(2):337–374, April 2000.

[6] J. H. Lin. “Divergence measures based on the shannon entropy ”. *IEEE Transactions on Information Theory*, 37:145–152, 1991.

[7] C. Liu and H. Y. Shum. Kullback-Leibler Boosting. In *Proc. IEEE 2003 Conf. on Compuer Vision and Pattern Recognition*, pages 407–411, May 2003.

[8] K. Messer, J. Kittler, M. Sadeghi, and M. Hamouz etc. “Face authentication test on the banca database”. In *Proceedings of International Conference Pattern Recognition*, pages 808–814, 2004.

[9] B. Moghaddam, C. Nastar, and A. Pentland. “A Bayesain similarity measure for direct image matching”. *Media Lab Tech Report* No. 393, MIT, August 1996.

[10] T. Ojala, M. Pietikainen, and D. Harwood. “A comparative study of texture measures with classification based on feature distributions”. *Pattern Recognition*, 29(1):51–59, January 1996.

[11] T. Ojala, M. Pietikainen, and M. Maenpaa. “Multiresolution gray-scale and rotation invariant texture classification width local binary patterns”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2002):971–987.

[12] R. E. Schapire and Y. Singer. “Improved boosting algorithms using confidence-rated predictions”. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pages 80–91, 1998.

[13] M.Pietikainen T.Ahonen, A. Hadid. “Face recognition with local binary patterns”. In *Proceedings of the European Conference on Computer Vision*, pages 469–481, 2004.

[14] I. Vajda. “Note on discrimination information and variation”. *IEEE Transactions on Information Theory*, 16:771–773, 1970.