

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

# INFORMATION TECHNOLOGY JOURNAL

**ANSI***net*

Asian Network for Scientific Information  
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

## Entropy-Directed AdaBoost Algorithm with NBBP Features for Face Detection

Hsun-Li Chang and Tai-Wen Yue

Tatung University/School of Computer Science and Engineering, Taipei, Taiwan

---

**Abstract:** AdaBoost learning algorithm had achieved good performance for real-time face detection with Haar-like features. Although the great achievement had been reached by AdaBoost, the learning phase is really time-consuming. This study introduces the so-called Neighboring-Block Binary Pattern (NBBP) features and associated each of them with a feature entropy to improve learning efficiency. By computing the entropies of NBBP features, the best weak-classifier of each iteration can be determined systematically in a non-brute-force manner. The concept is applied to build a real-time face detection system. Comparisons with other approaches will be presented in the study, including Receiver Operating Characteristic (ROC) and training efficiency. For still images, experimental results showed that the NBBP features are intrinsically superior to Haar-like ones in many perspectives, e.g., the former is illumination invariant and more efficacious in discriminate power. For 320×240 video sequences, we achieved the performance about 22 fps. In particular, the proposed feature selection methodology based on entropy provides a precise criterion to measure the discrimination ability of available features and, hence, can serve as a ‘referee’ for AdaBoost algorithm to construct weak classifiers effectively.

**Key words:** Face detection, AdaBoost, LBP, NBBP, NBBP entropy

---

### INTRODUCTION

The field of face detection has made significant progress in the past decade. There have been numerous reported approaches to face detection. Early works (before year 2003) had been nicely surveyed by Yang *et al.* (2002). Yang *et al.* (2002) divided the various approaches into four categories: knowledge-based methods, feature invariant approaches (Tareeq *et al.*, 2007), template matching methods (Park and Park, 2006) and appearance-based methods (Ishak *et al.*, 2006; Pakazad *et al.*, 2011). In general, appearance-based methods had been showing superior performance to the others. Recently, more attention is given to boosting-based face detection schemes which have evolved as the de-facto standard of face detection in real-world.

The real-time face detector based on AdaBoost algorithm proposed by Viola and Jones (2004) can be regarded as a milestone in face detection researches. The real-time performance was achieved by learning a sequence of Haar-like features. The Haar-like features encode differences in intensities between rectangles regions which can be calculated rapidly through the integral image. AdaBoost algorithm is used to select a small number of distinct weak classifiers from a large family to which they belong and combines these weak

classifiers as a strong classifier. Viola also adopted the cascaded structure to further speed up efficiency of face detection. Lienhart and Maydt (2002) used a set of rotated Haar-like features which significantly enrich the basic set of simple Haar-like features. The detector had better accuracy than Viola-Jones face detector. Mita *et al.* (2005) proposed joint Haar-like feature to capture the structural similarities within the face. It can be calculated very fast and has robustness against addition of noise and change in illumination. Yan *et al.* (2008) proposed the Locally Assembled Binary (LAB) Haar feature. Several neighboring binary Haar features were then assembled to capture their co-occurrence with similar idea to LBP. They showed that LAB feature was more efficient than Haar-like and LBP features both in discriminant power and computational cost. Because of the large set of features and the selecting procedure of weak classifiers, the AdaBoost training process of these features for face detection still takes a long time.

In this study, we propose the so-called Neighboring-Block Binary Pattern (NBBP for short) features and develop a feature selection strategy based on the entropy of NBBP features to improve the learning efficiency of AdaBoost algorithm. The idea is motivated from the information gain developed for Iterative Dichotomizer version 3 (ID3) (Quinlan, 1986; Christy and Thambidurai, 2006). Instead of exhaustively searching the feature with

minimal classification error in a brute-force manner, we compute the feature entropy for each NBBP and, then, choose the one with minimal entropy as the most capable one in classification straightforwardly. Comparing with Haar-like features, the total amount of competitive features for NBBP is much smaller. This provides us a factor to accelerate the learning process. Furthermore, NBBP feature is illumination invariant and more robust on reflecting the structure of the underlying patterns. This facilitates the face detection system to learn the true facial structure from training set with face and non-face images. The proposed feature selection methodology based on entropy can provide a precise criterion to measure the discrimination ability of available features and can serve as a ‘referee’ to improve the efficiency of AdaBoost algorithm.

### NEIGHBORING-BLOCK BINARY PATTERN FEATURES

Traditional Haar-like rectangle features measure the difference between the sum of the pixels within rectangular regions, these features can capture the intensity gradient at different locations by varying the position, size, shape or arrangement of rectangular regions. Viola and Jones (2004) applied three types of rectangle features for frontal face detection (Fig. 1). With the help of integral image, these features can be computed in constant time, so Haar-like features can achieve real-time detection. However, Haar-like features seem too simple and have some limits.

NBBP is a synonym of Multi-Block Local Binary Pattern (MB-LBP) proposed by Zhang *et al.* (2007). The basic idea to encode the NBBP features is to apply the LBP operator to neighboring rectangular blocks. Specifically, the original LBP operator was introduced by Ojala *et al.* (1996). It labels the pixels in a 3×3 image by thresholding the surrounding 8 pixels’ values with the center pixel’s value and then, encoding the results as a binary number. The NBBP extends the concept of LBP operator by applying it to 3×3 neighboring-blocks. Some possible layouts of 3×3 neighboring block is shown in Fig. 2 and the number inside each block denotes its corresponding index. All blocks have the same size and the value of each block is the sum of every pixel in each block which can be fast computed through integral image. The values of surrounding blocks will compare with the value of the central block and encode a non-metric value just like LBP.

We will use  $NBBP_j$  to denote the NBBP operator for the  $j^{th}$  nbbp feature. Its encoding scheme is described as:

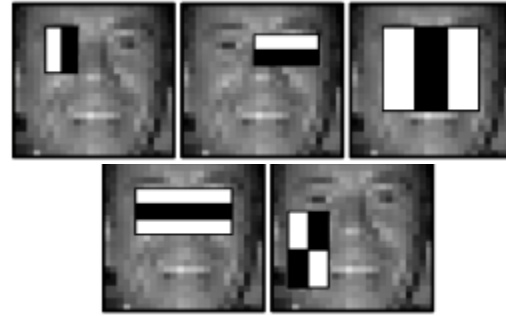


Fig. 1: Examples of Haar-like features

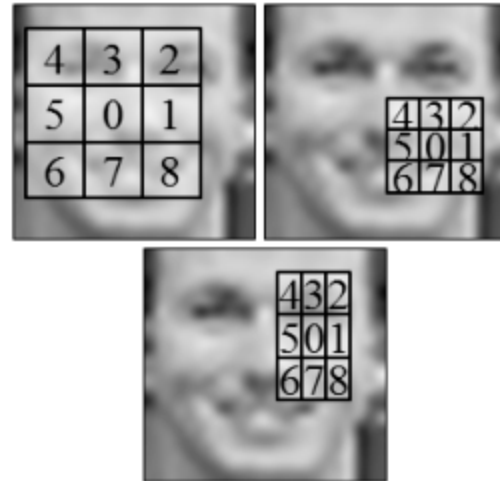


Fig. 2: Some possible layouts of NBBP features

$$NBBP_j(x) = \sum_{n=1}^8 s[\text{val}(x_n^j) - \text{val}(x_0^j)]2^{n-1}, \quad (1)$$

where,  $x^j$  represents sub-image cropped by the  $j^{th}$  NBBP,  $x_k^j$  represents the  $k^{th}$  block in  $x^j$  indexed using the scheme depicted in Fig. 2,  $\text{val}(x_k^j)$  represents the value of block  $x_k^j$  and:

$$s(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

An example of NBBP feature value encoding scheme is shown in Fig. 3.

The encoded binary value represents the feature value of an NBBP. Apparently, an NBBP feature covering larger area carry information for global texture and one with smaller area is for local texture. So NBBP features can

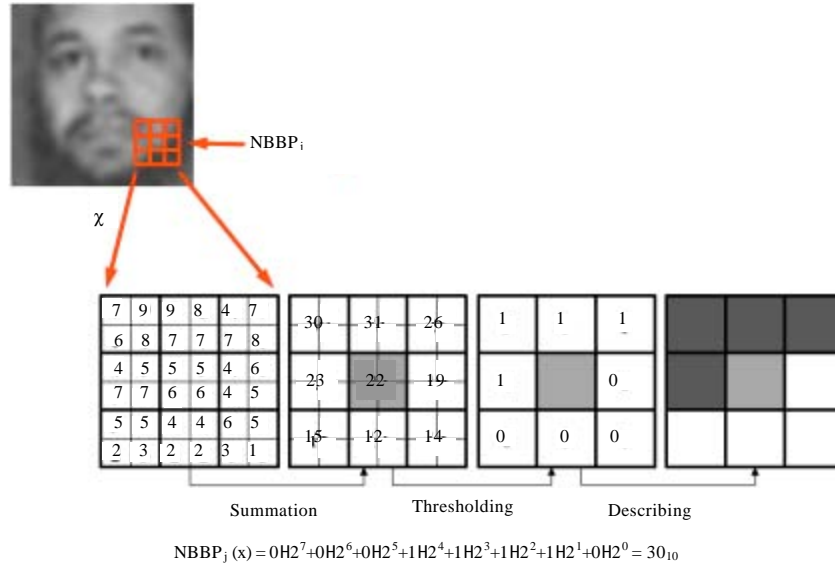


Fig. 3: A demonstration of NBBP feature value encoding scheme

detect diverse image structures such as edges, lines, spots, flat areas and corners. Besides, NBBP is encoded by comparison, so it doesn't need to do any lighting normalization. In other words, NBBP features are illumination invariant. Another advantage of NBBP feature is that the total number of NBBP-feature candidates is much smaller than the Haar-like ones. Considering an  $18 \times 18$  sub-window, for example, the ratio of the total number of NBBP candidates to the Haar-like ones is about 1 to 20 (2,601 : 51,705). This will greatly reduce the time needed in the feature selection stage of AdaBoost algorithm, to be detailed shortly. The experimental results from Zhang *et al.* (2007) also reveal that the NBBP features are more effective than Haar-like features and original LBP features.

### NBBP ENTROPY

In thermodynamics, entropy is the measurement of characterizing disorder, alias randomness (Laurendeau, 2005). Shannon (2001) connected such a measurement to the uncertainty of an information environment (Wang 2011). The common of these two is that the more uncertainty the system should possess, the larger the entropy. Pattern classification is aiming at categorizing things in order, i.e., less uncertainty. Therefore, machine-learning algorithm, e.g., AdaBoost, is, in fact, to seek some ways to minimize the system entropy effectively. In this section, we define the feature entropies of NBBP's associated to a given weighted training set, i.e., each training sample is assigned a advantage of the feature entropies to make a clever feature selection for AdaBoost Algorithm.

Consider a training set  $T = \{(x_i, y_i) : y_i \in \{-1, +1\}\} = P \cup N$ , where  $x_i$  represents the  $i$ th gray-level image,  $y_i = -1$  labels  $x_i$  as a negative example,  $y_i = +1$  labels  $x_i$  as a positive example and  $P$  and  $N$  partition  $T$  into the subsets containing all positive and negative examples, respectively. Furthermore, we assume that all images in  $T$  have the same size. Each training example is assigned a weight (probability density) which is defined by  $D = \{D(i)\}$  with  $D(i)$  being the associated weight of  $(x_i, y_i) \in T$ . Since  $D(i)$  is a probability density, it is very obviously that  $\sum_i D(i) = 1$ .

Let's now focus on the particular  $NBBP_j$ . Applied  $NBBP_j$  on  $P$  and  $N$  can obtain corresponding encoded feature values and two histogram  $H_j^+ = \{p_{j,0}^+, \dots, p_{j,255}^+\}$  and  $H_j^- = \{p_{j,0}^-, \dots, p_{j,255}^-\}$  associated with  $P$  and  $N$ , respectively, can be constructed by accumulating the probability densities associated with the examples in their corresponding training subsets (Fig. 4). Specifically, the elements in  $H_j^+$  and  $H_j^-$  can be obtained by:

$$p_{j,v}^+ = \sum_{(x_i, y_i) \in P} D(i) \cdot \delta(v, NBBP_j(x_i)) \quad (3)$$

$$p_{j,v}^- = \sum_{(x_i, y_i) \in N} D(i) \cdot \delta(v, NBBP_j(x_i)) \quad (4)$$

where,  $v = 0, \dots, 255$  and:

$$\delta(x, y) = \begin{cases} 1, & \text{if } x = y \\ 0, & \text{if } x \neq y \end{cases} \quad (5)$$

With  $H_j^+$  and  $H_j^-$  the (expected) entropy  $\epsilon_j$  for  $NBBP_j$  is defined by:

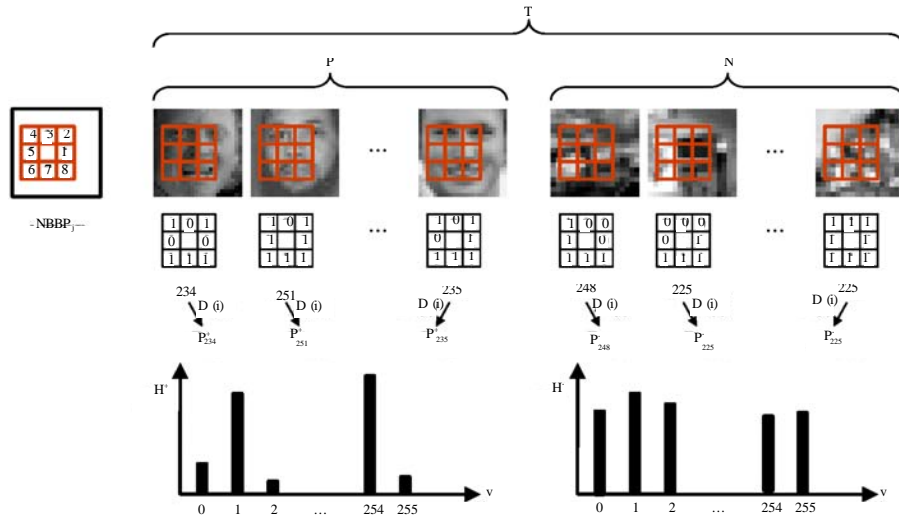


Fig. 4: NBBP-histogram construction from a training set

$$\epsilon_j = \sum_{v=0}^{255} p_{j,v} \left( -\frac{p_{j,v}^-}{p_{j,v}} \log_2 \frac{p_{j,v}^-}{p_{j,v}} - \frac{p_{j,v}^+}{p_{j,v}} \log_2 \frac{p_{j,v}^+}{p_{j,v}} \right) \quad (6)$$

$e_{j,v}$  : entropy for bin  $v$

where,  $p_{j,v} = p_{j,v}^+ + p_{j,v}^-$ . Based on the theory of entropy, one can easily see that the minimal and maximal possible values of  $\epsilon_j$  are 0 and 1, respectively.  $\epsilon_j = 0$  corresponds to the case that  $e_{j,v} = 0$  for all  $v = 0, \dots, 255$ , i.e.,  $p_{j,v}^- = 0$  and/or  $p_{j,v}^+ = 0$  for all  $v = 0, \dots, 255$ . This implies that each bin only contains patterns with single parity. Therefore, considering  $NBBP_j$  as a feature, it can separate the positive and negative examples in  $T$  completely. On the other hand,  $\epsilon_j = 1$  corresponds to the case that  $p_{j,v}^+ = p_{j,v}^-$  for all  $v = 0, \dots, 255$ . Such a case implies that  $NBBP_j$  is totally incapable for classification. The above observation reveals that the smaller the entropy for a feature is selected for classification, the less misclassification should be resulted.

### ENTROPY-DIRECTED ADABOOST ALGORITHM BASED ON NBBP

In this section, we describe the weak classifier based on NBBP-feature and summarize the improved version of AdaBoost algorithm which takes advantage of NBBP-feature entropy as a guidance for feature selection.

**Weak classifier construction:** Because the value of NBBP is non-metric, the weak classifier  $h_j(x)$  will not be in a threshold type like the one with Viola and Jones (2004).

A new type of weak classifier for NBBP features must be designed. Now, consider  $NBBP_j$  associated with histograms  $H_j^+ = \{p_{j,0}^+, \dots, p_{j,255}^+\}$  and  $H_j^- = \{p_{j,0}^-, \dots, p_{j,255}^-\}$ . The new type weak classifier  $h_j(x)$  based on multi-branch tree is defined by:

$$h_j(x) = \begin{cases} a_0, & NBBP_j(x) = 0 \\ \vdots \\ a_v, & NBBP_j(x) = v \\ \vdots \\ a_{255}, & NBBP_j(x) = 255 \end{cases} \quad (7)$$

and:

$$a_v = \begin{cases} +1, & \text{if } p_{j,v}^+ > p_{j,v}^- \\ -1, & \text{otherwise} \end{cases} \quad (8)$$

Therefore, given a pattern  $x$ , the output of  $h_j$  can be obtained by looking up the table associated to  $NBBP_j$ . Besides, this type of weak classifier is more compact, i.e., memory inexpensive, because each classifier needs only 32 bytes (256 bits) to store the table.

**Entropy-directed AdaBoost learning:** Basically, the training procedure of AdaBoost trains an optimal weak classifier, say,  $h_t$ , each iteration on weighted training samples, giving higher weight to samples that are currently misclassified. This is done  $T$  times, i.e.,  $t = 1, \dots, T$ , for a sequence of adaptively re-weighted samples. The final classifier, say,  $H$  is a linear combination of weak classifiers from each training stage. For comparison,

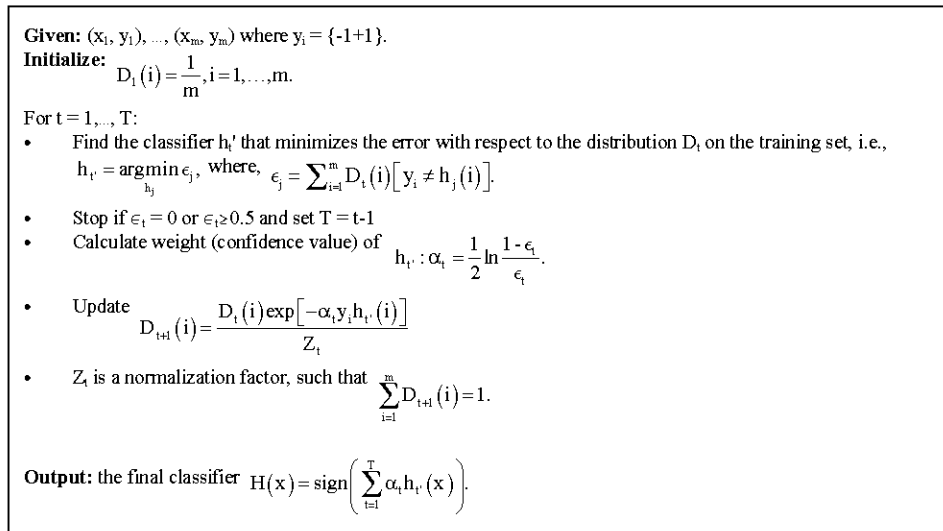


Fig. 5: Original discrete AdaBoost algorithm

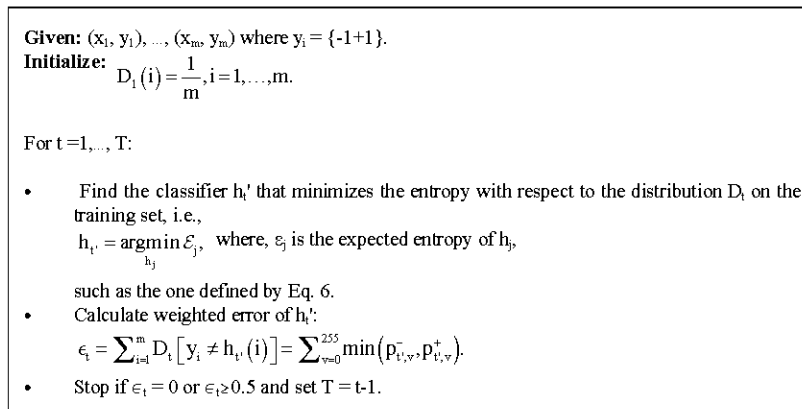


Fig. 6: Entropy-directed discrete AdaBoost algorithm

Fig. 5 and 6 list the original and the proposed Entropy-Directed AdaBoost algorithms, respectively. The shaded areas in the tables highlight the main difference of them.

The most time consuming and critical part of AdaBoost algorithm lies in selection of the optimal weak classifier from a huge number of candidates. The original AdaBoost algorithm finds the optimal weak classifier by determining the minimal errors of all possible weak classifiers in a brute force manner. A more efficient way to find the optimal weak classifier is to compute a reasonable index, such as the expected entropy defined by 6, to measure the optimality of each weak classifier. By comparing such indices, the most beneficial one will then come to the fore naturally. As shown in the shaded area of Fig. 6, present scheme finds the optimal weak classifier by simply evaluating the entropies of all possible

classifiers. The computation of the single weighted error  $\epsilon_t$  in Fig. 6 is simply for obtaining the weight  $\alpha_t$  of the currently selected classifier  $h_t'$  and for sample re-weighting.

We conduct a simple experiment to demonstrate that the classifier constructed by the proposed approach would yield better performance than Viola's approach. We used the popular CBCL Face Database to construct two 200-feature monolithic classifiers. All face and non-face images in the CBCL training set are used to train and the whole images in CBCL testing set are used to evaluate performance. The first one classifier was constructed by the Viola's approach with Haar-like features. The second one was constructed by the proposed Entropy-Directed Discrete AdaBoost algorithm with NBBP features. The comparative results ROC curve is shown in Fig. 7 and the training times is shown in Fig 8. According to the

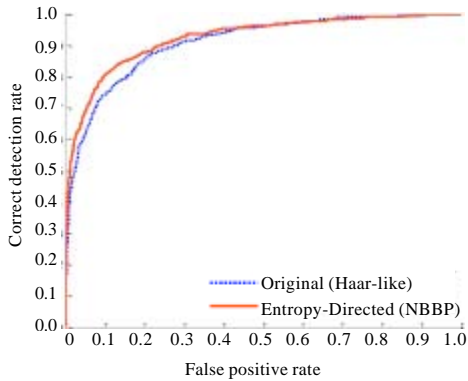


Fig. 7: Comparative ROC curve for the 200 feature classifiers

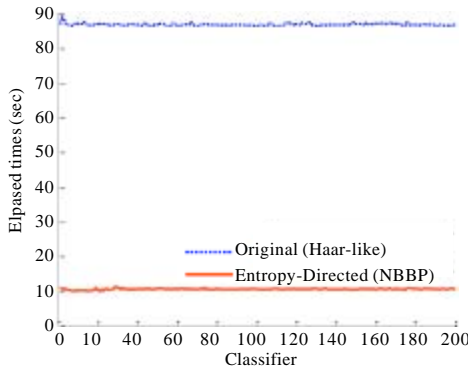


Fig. 8: The Comparative results for training time for the 200 feature classifiers

experimental results, the classifier based on the proposed approach with NBBP features are more effective and efficient than Viola's approach. It provides an initial evidence that the proposed scheme is an more effective technique for face detection.

**Face detection system by NBBP:** Although a single-stage face detection system can be built using the aforementioned methodology, it hardly achieves real-time requirement. Faces in an image possibly appear in arbitrary locations and with different scales. A face detection system needs repeatedly scanning the image thoroughly using different size of windows to fulfill its job. Therefore, such a windowing detector must evaluate tens of thousands of location/scale combinations just in order to classif few (e.g., 0 to 10) of them as faces in an image of moderate size. To achieve real-time, we thus need a mechanism being able to early rejection of obvious non-

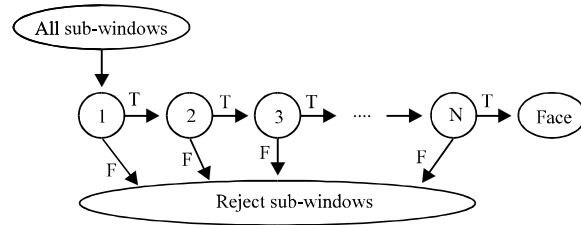


Fig. 9: Structure of cascaded face detector

face patterns. The cascaded structure and the training algorithm proposed by Viola and Jones (2004) had been successfully applied as a real-time face detector. A cascaded classifier has great performance in detection while radically reducing computation time. The overall detection process is a degenerate decision tree, what we call a “cascade” is schematically shown in Fig. 9. Our face-detection system adopts such a cascaded structure. Basically, we used the cascaded training algorithm described by Viola and Jones, (2004) to train our system except that the weak classifiers of each stage classifier in Fig. 9 are trained using the Entropy-Directed AdaBoost algorithm described in Fig. 6.

Our face dataset is composed of 24,000 face images obtained by mirroring 12,000 face images from LFW-a which is originated from LFW face database (Huang *et al.*, 2007). These faces in LFW-a were aligned using the commercial face alignment software, so there is no in-plane rotated face in our dataset. The face images in the dataset are rescaled to  $18 \times 18$  and partitioned to two subsets of equal size. One is for training and the other is for validation. Our non-face dataset is composed of 2,000 large images from bigfoto.com (<http://www.bigfoto.com>). Likewise, the dataset is divided into two subsets of equal size for training and validation, respectively. Training set is used to train the weak classifier while validation set is used to evaluate the detection rate and false positive rate of the trained cascaded classifier.

The 12,000 non-face examples used to train the first-stage classifier are collected by random sampling (of size  $18 \times 18$ ) sub-images from the 1,000 non-face images in the training set. The non-face examples used to train the classifiers of subsequent stages are the 12,000 false positives obtained by scanning the 1,000 non-face images in training set using the current cascaded classifier.

From the viewpoint of training, one difficulty is to collect the meaningful examples to represent non-faces since there are, in fact, infinite many non-face images. If the majority of non-face examples used for validation are obvious different from the face sample, the actual false positive rate for the resulting classifier may be

unacceptable. Therefore, we partition the learning process into two phases, say, the shallow learning phase and the deep learning phase, depending on the difficulty of non-face examples used for validation. Before learning, we randomly sample 1,000,000 non-face examples (of size  $18 \times 18$ ) for validation in the shallow learning phase from the 1,000 non-face images. After shallow learning phase completed, we use the partially completed cascaded classifier to exhaustively scan the 1,000 non-face images in the validation set to collect all false positive examples. In deep learning phase, we only use these latter examples for validation.

### EXPERIMENTAL RESULTS

The resulting cascaded face detector constructed using our proposed scheme has 16 stages with 481 weak classifiers in total. Figure 10 shows the development of detection rate and false positive rate of our validation data as the number of weak classifiers grows. In the Figure, the vertical solid lines in the Figure mark the start of the stage classifiers, the shaded area represents the period of shallow learning phase (the first 12 stages) and the non-shaded area (the last 4 stages) represents the period of deep learning phase.

To understand the entropy and error rate changes as the learning process proceeds, Fig. 11 shows the development of these two rates as the number of weak classifiers grows. The Figure reflects that the earlier stage-classifiers can achieve their subgoals (i.e., the assigned detection rate and false positive rate) rather easily. For example, the first 12 stages are made up by just only 194 classifiers. Furthermore, these two curves both tend to be flattened at their right ends. This reveals that the weak classifiers in the latter stages, in fact, make decision based on many non-common detail features. Figure 11 also implies that the lower expected entropy leads to lower weighted error. Hence, it is effective and reasonable to use the expected entropy of NBBP features to direct the feature selection process of AdaBoost algorithm.

An experiment was conducted using MIT/CMU test set which consists of 130 gray scale images with a total of 507 frontal faces. Figure 12 shows the output of our face detector on some test images from MIT/CMU test set. On a Intel Core 2 Quad 2.33GHz processor, our face detector can process  $320 \times 240$  video in about 22 fps. For comparison, we draw the ROC curves of our face detector and of the one trained using Gentle AdaBoost algorithm with MB-LBP as features on the same test set in Fig. 13.

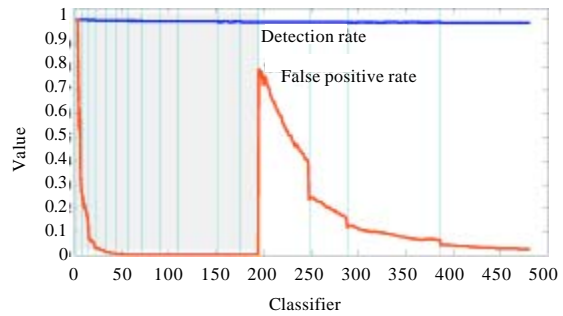


Fig. 10: The development of detection rate and false positive rate in training

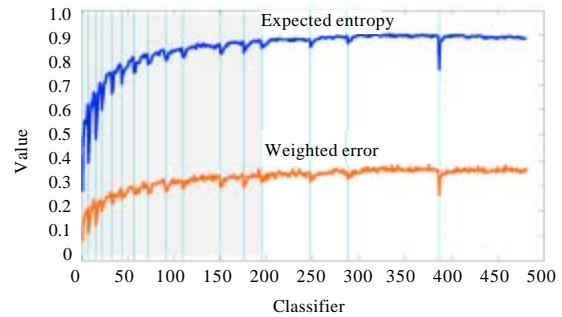


Fig. 11: The development of entropy and weighted error rate in training

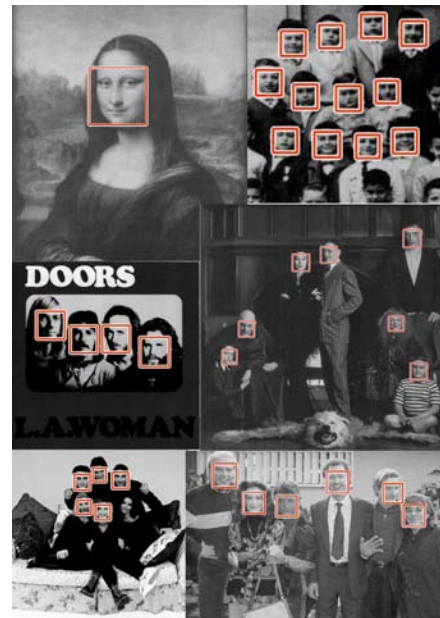


Fig. 12: Output of our face detector on a number of test images from the MIT/CMU test set



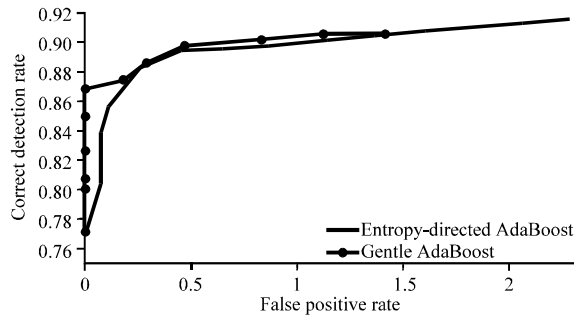


Fig. 13: ROC curves of two face detectors based-on the MIT/CMU test set

### DISCUSSION

As can be seen from the comparative results for the monolithic classifiers in Fig. 7, the proposed NBBP features had higher discriminant power than the Haar-like ones. It was caused by that NBBP features can detect diverse image structures. Furthermore, NBBP feature was illumination invariant and more robust on reflecting the structure of the underlying patterns. This facilitated the face detection system to learn the true facial structure from training set with face and non-face images. From Fig. 8, the proposed approach was more efficient than original one about 9 times. The most time consuming and critical part of AdaBoost algorithm lain in selection of the optimal weak classifier from a huge number of candidates. The higher efficiency of the proposed approach can be associated with two reasons. The first one was that the total amount of competitive features for NBBP is much smaller than Haar-like ones. Secondly, we used the expected entropy defined by Eq. 6, to measure the optimality of each weak classifier without a brute-force manner. Compared to the original Discrete AdaBoost algorithm, present scheme found the optimal weak classifier by simply evaluating the entropies of all possible classifiers. We also compared the performance of the proposed scheme with the Gentle AdaBoost based on MB-LBP features (Zhang *et al.*, 2007). Gentle AdaBoost used adaptive Newton steps for minimizing a weighed squared error at each step of feature selection process. According to Fig. 13, the Gentle AdaBoost had smaller false positive rate than the proposed scheme with detection rate lower than 88%. However, this detection rate was not acceptable for real-world applications. For higher detection rate, i.e., higher than 90%, present approach had better performance than the Gentle AdaBoost algorithm with MB-LBP features.

For building a real-time face detector by our approach, we used two phases learning scheme, say, the shallow learning phase and the deep learning phase, depending on the difficulty of non-face examples used for validation. As can be seen from Figure 10, it showed the development of detection rate and false positive rate of our validation data as the number of weak classifiers grows. In the Figure, the vertical solid lines in the Figure marked the start of the stage classifiers and the shaded area represented the period of shallow learning phase. The false positive rate approximated to zero rapidly after the 5th stage classifier. It represented that the resulting cascaded face detector constructed using present proposed scheme can achieve high detection rate and very low false positive rate. However, AdaBoost algorithm had the potential to overfit the training set because its objective is to minimize error on the training set (Bylaner and Tate, 2006). We used another validation set of non-face images to train our face detector after the shallow learning phase. At the beginning of deep learning phase, the non-shaded area in Fig. 10, the false positive rate was increasing obviously. It implied that our face detector was suffering from the overfitting problem. After the deep learning phase, our face detector can really reduce the actual false positive rate. So present face detector was more robust to apply on the real-world applications.

### CONCLUSIONS

In this study, we propose NBBP features for face detection and use entropy to guide the feature selection process of Discrete AdaBoost algorithm. This, as a result, can avoid a brute force approach to feature selection and, hence, can improve the learning efficiency to a certain degree. Present experiment justifies that our scheme works well both in learning efficiency and classification performance. Furthermore, the proposed concept can be easily applied to other AdaBoost relevant algorithms, e.g., Gentle AdaBoost, RealBoost and FloatBoost.

### ACKNOWLEDGMENTS

The authors would like to thank CMU for providing the face database.

### REFERENCES

- Bylaner, T. and L. Tate, 2006. Using validation sets to avoid overfitting in AdaBoost. Proceedings of the 19th International Florida Artificial Intelligence Research Society Conference, May 11-13, Florida, USA., pp: 544-549.

- Christy, A. and P. Thambidurai, 2006. Efficient information extraction using machine learning and classification using genetic and C4.8 algorithms. *Inform. Technol. J.*, 5: 1023-1027.
- Huang, G. B., M. Ramesh, T. Berg and E. Learned-Miller, 2007. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report. University of Massachusetts, Amherst, pp: 1-14. [http://tamaraberg.com/papers/Huang\\_eccv2008-lfw.pdf](http://tamaraberg.com/papers/Huang_eccv2008-lfw.pdf).
- Ishak, K.A., S.A. Samad and A. Hussain, 2006. A face detection and recognition system for intelligent vehicles. *Inform. Technol. J.*, 5: 507-515.
- Laurendeau, N.M., 2005. *Statistical Thermodynamics: Fundamentals and Applications*. Cambridge University Press, Cambridge, UK., ISBN-13: 9780521846356, pp: 448.
- Lienhart, R. and J. Maydt, 2002. An extended set of haar-like features for rapid object detection. *Int. Conf. Image Process.*, 1: 900-903.
- Mita, T., T. Kaneko and O. Hori, 2005. Joint haar-like features for face detection. *IEEE Int. Conf. Comput. Vision*, 2: 1619-1626.
- Ojala, T., M. Pietikainen and D. Harwood, 1996. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognit.*, 29: 51-59.
- Pakazad, S.K., F. Hajati and S.D. Farahani, 2011. A face detection framework based on selected face components. *J. Applied Sci.*, 11: 247-256.
- Park, C.W. and M. Park, 2006. Fast template-based face detection algorithm using quad-tree template. *J. Applied Sci.*, 6: 795-799.
- Quinlan, J.R., 1986. Induction of decision trees. *Mach. Learn.*, 1: 81-106.
- Shannon, C.E., 2001. A mathematical theory of communication. *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, 5: 3-55.
- Tareeq, S.M., R. Parveen, L.J. Rozario and M. Al-Amin-Bhuiyan, 2007. Robust face detection using genetic algorithm. *Inform. Technol. J.*, 6: 142-147.
- Viola, P. and M.J. Jones, 2004. Robust real-time face detection. *Int. J. Comput. Vision*, 57: 137-154.
- Wang, Y., 2011. Generalized Information theory: A review and outlook. *Inform. Technol. J.*, 10: 461-469.
- Yan, S., S. Shan, X. Chen and W. Gao, 2008. Locally Assembled Binary (LAB) feature with feature-centric cascade for fast and accurate face detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 24-26, Anchorage, Alaska, USA., pp: 1-7.
- Yang, M.H., D.J. Kreigman and N. Ahuja, 2002. Detecting faces in images: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24: 34-58.
- Zhang, L., R. Chu, S. Xiang, S. Liao and S. Li, 2007. Face detection based on Multi-Block LBP representation. *Adv. Biometrics*, 4642: 11-18.