Expression Recognition with Ri-HOG Cascade

Jinhui Chen[1], Zhaojie Luo[2], Tetsuya Takiguchi[2], and Yasuo Ariki[2]

[1]RIEB, Kobe University, Kobe, 657-8501, Japan
[2]Graduate School of System Informatics, Kobe University, Kobe, 657-8501, Japan

Abstract. This paper presents a novel classification framework derived from AdaBoost to classify facial expressions. The proposed framework adopts rotation-reversal invariant HOG as features. The Framework is implemented through configuring the Area under ROC curve (AUC) of the weak classifier with HOG, which is a discriminative classification framework. The proposed classification framework is evaluated with two very popular and representative public databases: MMI and AFEW. As a result, it outperforms the state-of-the-arts methods. This paper presents a novel classification framework derived from AdaBoost to classify facial expressions. The proposed framework adopts rotation-reversal invariant HOG as features. The Framework is implemented through configuring the Area under ROC curve (AUC) of the weak classifier with HOG, which is a discriminative classification framework. The proposed classification framework is evaluated with two very popular and representative public databases: MMI and AFEW. As a result, it outperforms the state-of-the-arts methods.

1 Introduction

Facial expression recognition (FER) is one of the most significant technologies for auto-analyzing human behavior. It can be widely applied to various application domains. Therefore, the need for this kind of technology in various different fields continues to propel related research forward every year.

In this paper, we propose a novel framework that adopts novel feature representation approach; namely, rotation-reversal invariant HOG (Ri-HOG) for learning boosting cascade. The proposed feature is reminiscent of Dalal *et al.*'s HOG [1], but the proposed feature representation approach noticeably enhances the conventional HOG-type descriptors for the image local features on invariant representation. For robustness and speed, we carry out a detailed study of the effects of various implementation choices in descriptor performance. We subdivide the local patch into annular spatial bins, to achieve spatial binning invariance. Besides, inspired by Takacs *et al.*'s rotation-invariant image features [2], we apply polar gradient to attaining gradient binning invariance, which is derived from the theory of polar coordinate. By doing so, the proposed method can significantly enhance the features descriptors in regard to invariant representation ability and feature descripting accuracy. Consequently, the proposed framework can robustly process out-of-plane head rotation cases.

The proposed learning model is derived from AdaBoost [3], but the proposed is implemented by configuring the area under the receiver-operating characteristic (ROC) curve (AUC) [4] to construct the weak classifier for expression classification. Adopting the AUC-based weak classifier, the false-positive-rate (FPR) of boosting training is adaptive to different stage, and it is usually much smaller than conventional approaches, which means its error rate is much smaller than the conventional approaches at each training iteration. Therefore, its convergence speed is much quicker than the conventional methods. Moreover, the accuracy of this classifier model is much better than conventional boosting classifiers.

We experimentally evaluated the proposed method in two public expression databases *i.e.*, MMI [5, 6], and AFEW [7], that together represent lab-controlled and real-world scenarios. The experimental results show that the proposed method can construct a robust FER system whose results outperform well-known state-of-the-art FER methods.

The main contribution of our study is the development of a novel framework, called Ri-HOG cascade, which can robustly process FER. In this paper, we are making the following original contributions: 1) we propose a robust local feature descriptor method called Ri-HOG, which is an appropriate similarity measure that can remain invariant for the rotated as well as reversed image representation; 2) We develop a novel cascade learning model that allows the FPR of boosting training is adaptive to different stage, in so doing, the convergence speed is quick and the accuracy of the classifiers is high.

The remainder of this paper is organized as follows: We describe the proposed framework in section 2. In section 3, we describe our experiments, and we draw our conclusions in section 4.

2 Proposed Method

Our proposed framework has these components: Ri-HOG features for local patch description; logistic regression-based weak classifiers, which are combined with AUC as a single criterion for cascade convergence testing; and a cascade for boosting training.

2.1 Feature Description

Background and problems: HOG is computed on a dense grid of uniformly-spaced cells and use overlapping local contrast normalization for improved accuracy. This feature is set based on *cells* and *blocks* representation system and it is widely used in classification applications, especially human detection. The describing ability of HOG features outperforms many existing features [8]. However, its robustness against image rotation is not satisfactory. Here one direct evidence is that the HOG feature is seldom applied to object tracking or image retrieval successfully. Giving a more scientific reason, see Fig. 1 for an example. Supposing Fig. 1(a) is an image with HOG block size, there are 4 cells in the

block. Fig. 1(b) is an image of Fig. 1(a) after making a quarter turn. HOG features are extracted from the two images individually. If the histogram of oriented gradients obtained from the regions 1, 2, 3, and 4 are severally denoted as x_1, x_2, x_3, x_4, then, the HOG features extracted from Fig. 1(a) and Fig. 1(b) are (x_1, x_2, x_3, x_4) and (x_3, x_1, x_4, x_2) respectively. This means that the rotation of image accompanies easily with the change of its HOG descriptors. Similarly, for reversal-image representation, HOG is also not invariant. Hence, we have to substantially enhance the robustness of HOG descriptors. Otherwise applications of HOG features would be limited to some narrow ranges.

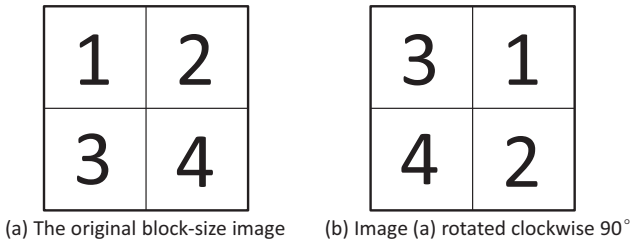

Fig. 1. Analyzing the robustness of conventional HOG descriptors in regard to image rotation.

Our approach: to robustly represent out-of-plane head rotation cases, we propose a novel feature descriptor on histograms of oriented gradients, *i.e.*, rotation-reversal invariant histograms of oriented gradients (Ri-HOG). We adopt annular spatial cells to replace rectangular cells (see Fig. 2(a)) and compute these cells on a dense polar gradient as feature descriptors. By doing so, the time complexity will not increase, but the invariant representation ability of the features will be extremely enhanced.

In this paper, we adopt polar gradient to represent the gradient for HOG descriptors, which is derived from Takacs *et al.*'s rotation-invariant image features [2]. But different from Takacs *et al.*'s approach, we only use the polar gradient to replace the Gaussian gradient function of conventional HOG. We subdivide the local patch into annular spatial cells (see Fig. 2(a)). How to calculate these descriptors is shown in Fig. 2. In Fig. 2(b), $\forall$ a point p in the circle c, the task is to compute the polar gradient magnitude of point p (x, y). Decompose the vector g into its local coordinate system as $(g^T r, g^T t)$, by projecting g into the r and t orientations as shown in Fig. 2(b). Since the component vectors of g in r and t orientations can be quickly obtained by $r = \frac{p-c}{\|p-c\|}$, $t = R_{\frac{\pi}{2}} r$, we can obtain the gradient g easily on the gradient filter. And, R_θ is the rotation matrix by angle θ.

Since Takacs *et al.* focus on image tracking applications, the speed is more important, they use Approximate Radial Gradient Transformation (ARGT) and ROC curve to compute the feature descriptors [2]. However, in this way, it will

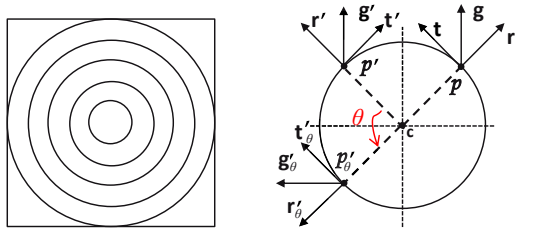

Fig. 2. Illustration of Ri-HOG descriptors.

decrease the distinctiveness of feature descriptors for recognition applications. In order to keep the distinctiveness of feature descriptors for recognition application, we do not follow Takacs *et al.*'s way to abandon gradient magnitudes, cells, and blocks representation system. Therefore, essentially, the feature (Ri-HOG) that we adopt here is an improved HOG feature, but the approach proposed by Takacs *et al.* is a very excellent and novel feature representation method for image tracking applications, which cannot be considered as a type of HOG feature. Ri-HOG persists and develops the discriminative representation of conventional HOG features. Meanwhile, it can also significantly enhances the descriptors with respect to rotation reversal invariant ability. Simply, we use the following four steps to extract the Ri-HOG descriptors:
1. Subdivide the local patch into annular spatial cells as shown in Fig. 2(a);
2. Calculate the polar gradient $(g^T r, g^T t)$ of each pixel in the cell;
3. Calculate the gradient magnitudes and the orientations of polar gradients using the Eq. 1:

$$\begin{aligned} M_{GRT}(x,y) &= \sqrt{(g^T r)^2 + (g^T t)^2}, \\ \theta(x,y) &= arctan\frac{g^T t}{g^T r}; \end{aligned} \tag{1}$$

4. Accumulating the gradient magnitude of polar gradient for each pixel over the annular spatial cells into 9 bins, which are separated according to the orientation of polar gradient. In this way, we can extract the feature descriptors from a dense annular spatial bin of these uniformly spaced cells.

Block normalization: We tried all of 4 normalization approaches listed by Dalal *et al.* in [1]. In practice, $L_2 - Hys$, L_2 normalization followed by clipping is shown working best. The recognition template is 100 × 100 with 10 cells, and it allows the patch size ranging from 50 × 50 pixels to 100 × 100 pixels. We slide the patch over the recognition template with 5 pixels forward to ensure enough feature-level difference. We further allow different aspect ratio for each patch (the ratio of width and height). The descriptors are extracted according to the

order from the inside to the outside of cells. Hence, concatenating descriptors in 10 cells together yield a 90-dimensional feature vector.

Now assume that the patch has been reversed and rotated by any given angle θ as shown in Fig. 2(b) (reversal: $p \to p'$; rotation: $p' \to p'_\theta$, the transformation orders can exchange). This yields a new local coordinate system and gradient: $p'_\theta = MR_\theta p,\ g'_\theta = MR_\theta g,\ r'_\theta = MR_\theta r,\ t'_\theta = MR_\theta t$, where M is the reversal matrix. As we known, the reversal matrix is a diagonal matrix with diagonal elements 1 or -1. Consequently, $M^T = M^{-1}$. The coordinates of the gradient in the local frame are invariant to reversal as well as rotation, which can be verified by

$$\begin{aligned} & (g'^T_\theta r'_\theta, g'^T_\theta t'_\theta) \\ = & ((MR_\theta g)^T MR_\theta r, (MR_\theta g)^T MR_\theta r) \\ = & (g^T R^T_\theta M^T MR_\theta r, g^T R^T_\theta M^T MR_\theta t) \\ = & (g^T r, g^T t). \end{aligned} \tag{2}$$

Since the point $p(x, y)$ as well as the angle θ are any given ones, and all gradients are transformed via the same way; *i.e.*, they are one-to-one mapping. Thus, the set of gradients on any given point around the patch is invariant to reversal as well as rotation.

2.2 Training Weak Classifier

In this study, we build a weak classifier over each local patch described by the Ri-HOG descriptor, and select the optimum patches in each boosting iteration from the patch pool. Meanwhile, we construct the weak classifier for each local patch by logistic regression to fit our classifying framework, due to it being a probabilistic linear classifier.

On one hand, we build a weak classifier over each local patch, as described by the descriptor, and select optimum patches in each boosting iteration from the patch pool. On the other hand, we construct a weak classifier for each local patch by logistic regression to fit our classification framework, since it is a probabilistic linear classifier. Given a Ri-HOG feature $\mathbb{F}$ over a local patch, logistic regression defines the probability model:

$$P(q|\mathbb{F}, \mathbf{w}) = \frac{1}{1 + \exp(-q(\mathbf{w}^T \mathbb{F} + b))}, \tag{3}$$

when $q = 1$ means that the trained sample is a positive sample of the current class, $q = -1$ indicates negative samples, $\mathbf{w}$ is a weight vector for the model, and b is a bias term. We train classifiers on local patches from a large-scale dataset. Assuming, in each boosting iteration stage, that there are K possible local patches, which are represented by Ri-HOG feature $\mathbb{F}$, each stage is a boosting training procedure with logistic regression as weak classifiers. In this way,

the parameters can be identified by minimizing the objective:

$$\sum_{k=1}^{K} \log(1 + \exp(-q_k(\mathbf{w}^T \mathbb{F}_k + b))) + \lambda \left\|\mathbf{w}\right\|_p, \tag{4}$$

where λ denotes a tunable parameter for the regularization term, and $\left\|\mathbf{w}\right\|_p$ is the L_p norm of the weight vector. Note that it is also applied to L_2-loss and L_1-loss linear support vector machines (SVMs) by the well-known open source code LIBLINEAR [9]. Therefore, this question can be solved using algorithms in [9]. In this study, the weak classifier is defined as:

$$h(\mathbb{F}) = 2P(q|\mathbb{F}, \mathbf{w}) - 1. \tag{5}$$

We trained the boosting cascade on local patches from a large-scale dataset. In practice, AdaBoost is not skilled at processing the vector-descriptor feature directly. Inspired by Li *et al.*'s SURF cascade [10], we found that the AUC score [11] can solve the problem. Therefore, by innovating the AUC score, we can avoid the difficult convergence risk.

Given the weak classifiers h_n for cascade iteration n, the strong classifier is defined as $H_N(\mathbb{F}) = \frac{1}{N}\sum_{n=1}^{N} h_n(\mathbb{F})$. Assuming there are a total of N boosting iteration rounds, in the round n, we will build K weak classifiers $[h_n(\mathbb{F}_k)]_{k=1}^{K}$ for each local patch in parallel from the boosting sample subset. Meanwhile, we also test each model $h_n(\mathbb{F}_k)$ in combination with previous $n-1$ boosting rounds. In other words, we test $H_{n-1}(\mathbb{F})+h_n(\mathbb{F}_k)$ for $H_n(\mathbb{F})$ on the all training samples, and each test model will produce a highest AUC score [4, 11] $J(H_{n-1}(\mathbb{F}) + h_n(\mathbb{F}_k))$. *i.e.*,

$$S_n = \max_{k=1,\cdots K} J(H_{n-1}(\mathbb{F}) + h_n(\mathbb{F}_k)). \tag{6}$$

This procedure is repeated until the AUC scores converge, or the designated number of iterations N is reached.

The whole procedure involves a forward selection and inclusion of a weak classifier over possible local patch temples that can be adjusted using different temple configurations, according to the processing images. To enhance both the speed of learning convergence and robustness, our algorithm further introduces a backward removal approach. For more details on including backward removal or even a floating searching capability into the boosting framework, please refer to [12]. In this study, we implement backward removal on Algorithm 1 step 4, to extend the procedure with the capability to backward remove redundant weak classifiers. In so doing, it is not only able to reduce the number of weak classifiers in each stage, but also able to improve the generalization capability of the strong classifiers. The details of how to implement these learning approaches are indicated in Algorithm 1.

Boosting Cascade Training To the best of our knowledge, almost all existing cascade detection frameworks are trained based on two conflicting criteria, *i.e.* the false-positive-rate (FPR) f_j and the hit-rate (or recognition rate) r_j for the

Algorithm 1 Learning Boosting Classifiers.

Require:

1. Given: the number of label categories M and the overall sample set $\mathbf{S} = \{(x_1, y_1), \cdots, (x_\tau, y_\tau)\}$, where τ is the number of the samples;
2. Initialize the weight parameter w_0 for positive (labeled as "+") samples and negative (labeled as "-") samples:
   a. $w_0^+ = 1/(M \times \tau_+)$ for those $q = 1$;
   b. $w_0^- = 1/(M \times \tau_-)$ for those $q = 1$;
3. **for** $(n = 0; n < N; n = n + 1)$ **do**
   a. Sampled $30 \times p$ (in this paper, $p = 3$) positive samples and $30 \times p$ negative samples from training set;
   b. Parallel replace each Ri-HOG template to train a series of logistic regression models $[h_n(\mathbb{F}_k)]_{k=1}^K$;
   c. In order to obtain the AUC score, calculate $H_{n-1}(\mathbb{F}) + h_n(\mathbb{F}_k)$ on the best model of previous stage: S_{n-1} and each $h_n(\mathbb{F}_k)$;
   d. Choose the best model S_n which contains the best weak classifier $h_n(\mathbb{F}_k)$, according to the Eq. 6;
   f. Update weight

$$w_{n+1} = \frac{w_n \exp(-q_n h_n(\mathbb{F}_k))}{Z_n},$$

   where Z_j is a normalization factor, on which it can make the weight follow to $M \sum w^+ = 1$ and $M \sum w^- = 1$;
   g. If AUC value S_n is converged, break the loop;
   end for
4. In order to ensure the overall AUC score to be the highest one, test all learned models during the current iteration process:
   for $(k = 0; k < K; k = k + 1)$ **do**
   if $H_{n-1}(\mathbb{F}) + h_n(\mathbb{F}_k) > S_n$ **then**
   a. $S_n = H_{n-1}(\mathbb{F}) + h_j(\mathbb{F}_k)$;
   b. Empty those unnecessary data to free the memory;
   end if
   end for
5. Output final strong model H_N for this stage.

detection-error tradeoff. The overall FPR of a $T-$stage cascade is $F = \prod_{j=1}^{T} f_j$, while the overall hit-rate is $R = \prod_{j=1}^{T} r_j$. Usually, the maximum suggested setting of f_j is 0.5 [13]. Therefore, for the system to reach an overall FPR= 10^{-6}, it requires at least 20 stages ($0.5^{20} \approx 10^{-6}$) by the given global setting. Note that some stages may reach this goal without convergence. Hence, it is better that the FRP be adaptive among different stages so that we could easily reach the overall training goal. Some automatic scheme methods [14–17] tune the intermediate thresholds of each stage. These approaches may alleviate painful manual tuning efforts, but do not address the convergence speed. Therefore, we do not consider these appropriate for implementing our cascade-type ensemble of weak classifiers.

Inspired by [4] and [10], here we introduce AUC as a single criterion for cascade convergence testing, which realizes an adaptive FPR among different stages. Hence, combined with logistic regression-based weak classifiers to adopt Ri-HOG features, this approach can yield a fast convergence speed and a cascade model with much shorter stages.

Algorithm 2 Training Multithreaded Boosting Cascade

Require:
1. Over all FPR: F_N for $i-$th category data;
2. Minimum hit-rate per stage $d_i^{(min)}$;
3. Current class samples: $\mathbf{X}_i^+$;
4. Non-current class samples: $\mathbf{X}_i^-$;
5. The number of expression labels: M;

Initialize: $j = 0, F_i^{(j)} = 1, D_i^{(j)} = 1$;
for $(i = 0; i < M; i = i + 1)$ **do**
  while $(F_i^{(j)} > F_i^{(n)})$ **do**
    1. j=j+1;
    2. Train a stage classifier $H_i^{(j)}(\mathbb{F})$ by samples of $\mathbf{X}^+$ and $\mathbf{X}^-$ via Algorithm 1;
    3. Evaluate the model $H_i^{(j)}(\mathbb{F})$ on the whole training set to obtain ROC curve;
    4. Determine the threshold $\theta_i^{(j)}$ by searching on the ROC curve to find the point $(d_i^{(j)}, f_i^{(j)})$ such that $d_i^j = d_i^{(min)}$, but when existing the mimimum one $d_i^{(j)}$ that follows to the condition: $d_i^{(j)} < d_i^{(min)}$, set $d_i^{(min)} = d_i^{(j)}$ to update the minimal hit-rate;
    5. Update: $F_i^{(j)} = F_i^{(j-1)} \times f_i^{(j)}$,
    $D_i^{(j)} = D_i^{(j-1)} \times d_i^{(j)}$;
    6. Empty the set $\mathbf{X}_i^-$;
    7. **while** $(F_i^{(j)} > F_i^{(j-1)}$ and size $|\mathbf{X}_i^+| \neq |\mathbf{X}_i^-|\ )$ **do**
    Adopt current cascade detector to scan non-target images with sliding window and put false-positive samples into $\mathbf{X}_i^-$;
    end while
  end while
end for
8. Output the boosting cascade detector $\{H_i^{(j)} > \theta_i^{(j)}\}$ and overall training accuracy F and D.

To avoid overfitting, we restricted the number of samples used during training, as in [14]. In practice, we sampled an active subset from the whole training set according to the boosting weight. It is generally good practice to use about $30 \times p$ samples of each class, where p is a multiple coefficient (Algorithm 1 step 3.a).

Within one stage, no threshold for intermediate weak classifiers is required. We need only determine each decision threshold θ_i. In our case, using the ROC curve, the FPR of each emotional category is easily determined when given the minimal hit-rate $d_i^{(min)}$. We decrease $d_i^{(j)}$ from 1 on the ROC curve, until reaching the transit point $d_i^j = d_i^{(min)}$. The corresponding threshold at that point is the desired θ_i, *i.e.*, the FPR is adaptive to different stage, and it is usually much smaller than 0.5.

After one stage of classifiers learning is converged via Algorithm 2, we continue to train another one with false-positive samples coming from the scanning of non-target images with the partially trained cascade . We repeat this procedure until the overall FPR reaches the stated goal. In ding so, the FPR is usually much smaller than 0.5 and it is adaptive for different stages. Therefore, this approach can result in a model size that is much smaller, and has the recognition speed and accuracy that is dramatically increased.

3 Experiments

In this section, we provide details of the dataset and evaluation results for the proposed method. We implemented all training and recognition programs in C++ on Win 10 OS, processed with a PC with a Core i7-6700K 4.0 GHz CPU and 32 GB RAM.

3.1 Databases and Protocols

We evaluated the proposed method on two reference databases, *i.e.* MMI, and AFEW, which include the lab-controlled database and the database in the wild.
MMI DB The MMI DB is a public database that includes more than 30 subjects, in which the female-male ratio is roughly 11:15. The subjects' ages range from 19 to 62, and they are of European, Asian or South American descent. This database is considered to be more challenging than CK+ [18], because there are many side-view images and some posers have worn accessories such as glasses. To evaluate the out-of-plane head rotation cases clearly, we adopt MMI database representing the lab-controlled to test proposed approaches in this paper. In the experiments, we used all 205 effective image sequences of the six expressions in the MMI dataset. In the recognition stage, the images of MMI were made into videos according to the person-independent.
AFEW DB For the AFEW DB, which is a much more challenging database, evaluation experiments also have been done [7]. All of the AFEW sets were collected from movies to depict so-call wild scenarios. In experiments, the videos in training set are decomposed into images for training. We trained the training

set and the results are reported for its validation set, in the same way as for the latest FER work [19].

We used all training samples in AFEW training set and collected training samples from according to the person-independent 10-fold cross-validation rule. In order to reduce the process time of training, the samples from two datasets were trained together. All of training samples were normalized to 100×100-pixel facial patches. In order to enhance the generalization performance of boosting learning, we dealt with the training samples by some transformations (mirror reflection, rotate the images *etc.*), finally, the original samples were increased by a factor of 64. The testing sample sequences were not done on any normalization. In the training stages, the training data of current processing expression were adopted as positive sample data; the other expressions' data were used for negative data.

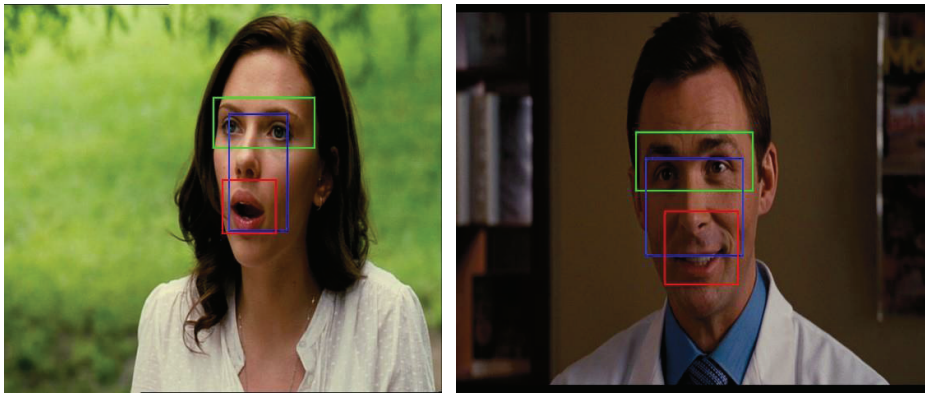

Fig. 3. Top-3 local patches picked by training procedure in the green-red-blue order on AFEW database.

3.2 Training Speed Evaluation Results

We replaced 40 types of the local patches on the 100×100 detection template as described in subsection 2.1. The proposed method used 377 minutes to converge at the $16th$ iteration stage. The cascade detector contained $2,394$ classifiers of all categories, and only need to evaluate 1.5 HOG per window. After training, we observed that the top-3 picked local patches for FER laid in the regions of two eyes and mouth. This situation is similar to Haar-based classifiers [20], see the examples in Fig. 3.

More details for cascade of FER are illustrated in Fig. 4(a) and Fig. 4(b), which include the number of weak learners in each stage and the average accu-

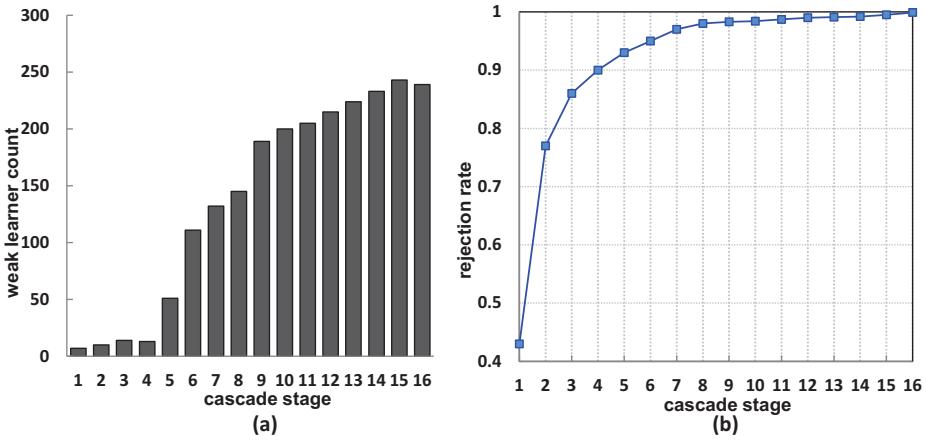

Fig. 4. (a) The number of weak classifiers at each cascade stage; (b) the accumulated rejection rate over all stages.

mulated rejection rate over the whole cascade stages. It shows that the first 8 stages have rejected 98% of the non-current class samples.

In order to evaluate the convergence speed of the AUC model, we determined the FPR at each boosting stage. The results show that, in the AUC model, the FPR f_j at each cascade stage is adaptive among the different stages , ranging from 0.04101 to 0.22337, is much smaller than the conventional model FPR of 0.5. In almost all existing cascade frameworks FPR $\prod_{j=1}^{T} f_j$ (T denotes the total cascade stages) reaches the goal (It is usually set as 10^{-6}). This means that conventional models require more iterations and that the AUC model cascade can converge much faster. These relate directly to training efficiency and recognition speed. Therefore, these experimental results confirm that the AUC cascade model is much more efficient than conventional cascade models. However, since the proposed framework makes the classifiers parallel recognize the multiclass expressions, the peak of memory cost is nearly six times more than the conventional one.

3.3 Recognition Results Comparison

In this paper, all the labels of the expression categories were named the same as they are in the original databases. Since the proposed is a binary classification framework, we test the expression class one by one. The facial region is detected by V-J framework [3] implemented by Open CV, and expression in face is recognized on proposed approaches. Here we show the recognition results on MMI, and AFEW.

Adopting Ri-HOG features, we evaluated almost of existing classifiers proposed for cascade learning and top ones of them are reported in Table 2. The

results show that the proposed classifier is more suitable for processing FER. The reason why we have to adopt Ri-HOG as features is also shown in Table 2; *i.e*, it dominates others on the accuracy. Meanwhile, its recognition speed can meet the real-time recognition. However, adopting SIFT as features, the real-time recognition is an impossible task (speed: only 18 frames per second), although the performance of the proposed framework with SIFT is also quite excellent.

Table 1. Recognition results on MMI and AFEW.

Method	Accuracy on MMI (%)							Accuracy on AFEW(%)						
	An	Di	Fe	Ha	Sa	Su	Ave.	An	Di	Fe	Ha	Sa	Su	Ave.
HOE [21]	46.4	58.3	33.2	62.6	60.8	65.1	55.5	11.2	16.5	9.0	33.5	15.3	28.3	19.0
LBP-TOP [22]	58.1	56.3	53.6	78.6	46.9	50.0	57.2	11.7	**19.6**	17.9	42.3	**23.8**	33.6	24.8
HOG 3D [23]	61.3	53.1	39.3	78.6	43.8	55.0	55.2	–	–	–	–	–	–	26.9
ITBN [24]	46.9	54.8	57.1	71.4	65.6	62.5	59.7	91.1	94.0	83.3	89.8	76.0	91.3	86.3
LSH [25]	59.6	**71.4**	62.3	68.9	**70.3**	75.1	61.8	23.1	12.8	**38.6**	9.7	21.1	10.9	19.4
3D LUT [20]	43.3	55.3	56.8	71.4	28.2	77.5	47.2	45.7	0	0	**62.0**	13.2	48.6	28.2
3DCNN-DAP [26]	64.5	62.5	50.0	**85.7**	53.1	57.5	62.2	–	–	–	–	–	–	–
STM [19] –	–	–	–	–	–	–	65.4	–	–	–	–	–	–	31.7
Baseline [7]	–	–	–	–	–	–	–	50.0	25.0	15.2	57.1	16.4	21.7	33.2*
Ours	**70.2**	60.4	**76.5**	81.2	62.1	**84.2**	**72.4**	**56.2**	36.3	**48.5**	**74.6**	**36.0**	**89.1**	**56.8**

The comparison moths were selected to represent the state-of-the-art level of this field, which includes proposing for the improvement of local spatiotemporal descriptors: such as LBP-TOP [22], HOE [21], HOG 3D [23], which are very popular for FER, while 3DCNN-DAP [26] and STM [19] are the latest ones; also including those methods that focus on enhancing the robustness of their classifying frameworks or making the frameworks can be encoded robustly, like, ITBN [24], 3D LUT [20] and LSH-CORF [25] *etc.* For fair comparison with them, we used the same databases, which were evaluated via the standardized items what they had done.

Table 1 compares our method with these state-of-the-art methods. Furthermore, almost of these meothods were conducted using their released codes and the parameters had been tuned to better-adapt for our experiments. However, about some methods, because we cannot obtain their source codes until now (*e.g.* STM [19] and 3DCNN-DAP [26], *etc.*), thus, we have to cite the reported results from the related works. The precisions of our framework (Ri-HOG cascade) were 72.4% on MMI database, and 56.8% on AFEW whose baseline is 30.9% (*the results are cited from the work[7], yet we donot test the Neutral class in this paper). The state-of-the-art levels were improved 7% and 25.1% respectively by the proposed framework on MMI and AFEW. In addition, the recognition speed of the proposed framework reached 55 frames per second.

4 Conclusion

In this paper, we have proposed a novel cascade framework called rotation-reversal invariant HOG cascade for robust FER. The proposed framework adopts

Table 2. Average precision using different classifiers and features.

Database	Precision of classifiers (%)				Precision of feature (%)				
	BinBoost [27]	JC [28]	SC [15]	Proposed	SIFT	SURF	Haar	HOG	Ri-HOG
MMI	62.6	55.9	50.2	**72.4**	65.4	46.0	42.2	58.8	**72.4**
AFEW	43.9	40.6	26.8	**56.8**	41.5	35.8	17.3	32.4	**56.8**

Ri-HOG for robustly process out-of-plane head rotation cases. Meanwhile, in the cascade learning, the proposed method use AUC as a single criterion for cascade convergence testing to enhance the classifiers learning. We used two representative public databases in FER research field, to experimentally confirm the validity of the proposed method. These issues are important to those with related research interests.

About the future work, we will attempt to study the question about how does the feature representation error impact on recognition frameworks.

References

1. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Hetection. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR). (2005) 886–893 vol. 1
2. Takacs, G., Chandrasekhar, V., Tsai, S., Chen, D., Grzeszczuk, R., Girod, B.: Fast computation of rotation-invariant image features by an approximate radial gradient transform. IEEE Trans. Image Proc.(TIP) **22** (2013) 2970–2982
3. Viola, P., Jones, M.: Robust Real-Time Face Detection. Int. J. Comput. Vis. (IJCV) **57** (2004) 137–154
4. Ferri, C., Flach, P.A., Hernández-Orallo, J.: Learning Decision Trees Using the Area Under the ROC Curve. In: Proc. Int. Conf. Machine Learn. (ICML). (2002) 139–146
5. Valstar, M.F., Pantic, M.: Induced Disgust, Happiness and Surprise: an Addition to the MMI Facial Expression Database. In: Proc. of Int. Conf. Language Resources and Evaluation, Workshop on EMOTION. (2010) 65–70
6. Pantic, M., Valstar, M.F., Rademaker, R., Maat, L.: Web-based Database for Facial Expression Analysis. In: Proc. IEEE Int. Conf. on Multimedia and Expo (ICME). (2005) 317–321
7. Dhall, A., Goecke, R., Lucey, S., Gedeon, T.: Collecting Large, Richly Annotated Facial-Expression Databases from Movies. MultiMedia, IEEE **19** (2012) 34–41
8. Thurau, C., Hlavac, V.: Pose Primitive Based Human Action Recognition in Videos or Still Images. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR). (2008) 1–8
9. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A Library for Large Linear Classification. J. Mach. Learn. Res. **9** (2008) 1871–1874
10. Li, J., Wang, T., Zhang, Y.: Face Detection Using SURF Cascade. In: Proc. IEEE Int. Conf. Comput. Vis. (ICCV) Workshops. (2011) 2183–2190
11. Long, P., Servedio, R.: Boosting the Area under the ROC Curve. In: Proc. Adv. Neural Inf. Proc. Syst. (NIPS). (2007) 945–952
12. Li, S.Z., Zhang, Z., Shum, H.Y., Zhang, H.: FloatBoost Learning for Classification. In: Proc. Adv. Neural Inf. Proc. Syst. (NIPS). (2002) 993–1000

13. Li, J., Zhang, Y.: Learning SURF Cascade for Fast and Accurate Object Detection. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR). (2013) 3468–3475
14. Xiao, R., Zhu, H., Sun, H., Tang, X.: Dynamic Cascades for Face Detection. In: Proc. IEEE Int. Conf. Comput. Vis. (ICCV). (2007) 1–8
15. Bourdev, L., Brandt, J.: Robust Object Detection via Soft Cascade. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR). Volume 2. (2005) 236–243
16. Sochman, J., Matas, J.: WaldBoost - Learning for Time Constrained Sequential Detection. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR). Volume 2. (2005) 150–156
17. Brubaker, S., Wu, J., Sun, J., Mullin, M., Rehg, J.: On the Design of Cascades of Boosted Ensembles for Face Detection. Int. J. Comput. Vis. (IJCV) **77** (2008) 65–86
18. Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops. (2010) 94–101
19. Liu, M., Shan, S., Wang, R., Chen, X.: Learning Expressionlets on Spatio-temporal Manifold for Dynamic Facial Expression Recognition. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR). (2014) 1749–1756
20. Chen, J., Ariki, Y., Takiguchi, T.: Robust Facial Expressions Recognition Using 3 D Average Face and Ameliorated Adaboost. In: Proc. ACM Multimedia Conf. (MM). (2013) 661–664
21. Wang, L., Qiao, Y., Tang, X.: Motionlets: Mid-level 3D Parts for Human Motion Recognition. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR). (2013) 2674–2681
22. Zhao, G., Pietikainen, M.: Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions. IEEE Trans. Patt. Analy. Mach. Intell. (TPAMI) **29** (2007) 915–928
23. Klaeser, A., Marszalek, M., Schmid, C.: A Spatio-temporal Descriptor Based on 3D-gradients. In: Proc. British Machine Vis. Conf. (BMVC). (2008) 99.1–99.10
24. Scovanner, P., Ali, S., Shah, M.: A 3-dimensional Sift Descriptor and Its Application to Action Recognition. In: Proc. ACM Multimedia Conf. (MM). (2007) 357–360
25. Rudovic, O., Pavlovic, V., Pantic, M.: Multi-output Laplacian Dynamic Ordinal Regression for Facial Expression Recognition and Intensity Estimation. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR). (2012) 2634–2641
26. Liu, M., Li, S., Shan, S., Wang, R., Chen, X.: Deeply Learning Deformable Facial Action Parts Model for Dynamic Expression Analysis. In: Proc. Asia Conf. Comput. Vis. (ACCV). Volume 9006. (2014) 143–157
27. Trzcinski, T., Christoudias, M., Lepetit, V.: Learning Image Descriptors with Boosting. IEEE Trans. Patt. Analy. Mach. Intell. (TPAMI) **37** (2015) 597–610
28. Chen, D., Ren, S., Wei, Y., Cao, X., Sun, J.: Joint Cascade Face Detection and Alignment. In: Proc. Eur. Conf. Comput. Vis. (ECCV). (2014) 109–122