Vol.2016-CG-165 No.1
Vol.2016-DCC-14 No.1
Vol.2016-CVIM-204 No.1
2016/11/9

Mixed Features for Face Detection in Thermal Image

Chao Ma[1,a)] Ngo Thanh Trung[1,b)] Hideaki Uchiyama[1] Hajime Nagahara[1] Atsushi Shimada[1] Rin-ichiro Taniguchi[1]

Abstract: An infrared camera is able to capture temperature distribution as an infrared (IR) image. It is a powerful tool in human related applications, such as human face recognition in complex illumination and fever screening in public places relying on facial temperature. Since facial temperature is almost constant, it is easy to find the facial region on an IR image. However, a simple temperature thresholding is not always working for detecting face stably. It is a standard for face detection to use Adaboost with local features such as Haar-like, MB-LPB, and HoG in visible image. However, there are very few research works using these local features in IR domain. In this paper, we propose an AdaBoost based training method to mix these local features for face detection in IR domain. In experiment, we captured a dataset of 20 people including 14 males and 6 females with variations of 10 different distances, 21 poses, and with/without glasses. We showed the proposed mixed features has an advantage over all of the regular local features using leave-one-out cross-validation.

1. Introduction

Camera technology has been developed for many decades, with different categories of cameras we can record images in different spectrums. A thermal imager is used to capture an image in long wave infrared spectrum. Since it is possible to see the temperature of objects, it is a powerful tool in industry inspection, medical imaging, chemical imaging, surveillance, etc.

We can also discriminate a face from an infrared image because the temperature of the facial region is stably around 37 °C.

There are variety of face detection methods in thermal spectrum which can be divided into three main categories: segmentation based, projection based, and machine learning based methods. Segmentation based method [1] is the most straightforward way relying on constant temperature of human face. By assuming that face temperature is within a range around 37 °C, we can simply threshold the image to find the facial region. Projection based method [2], [3] is a more complex approach which assumes that the temperature of facial region is higher than that of the background and finds the locations of facial region from the vertical and horizontal projection profiles. Machine learning based method takes face and non-face patches as positive and negative samples, and employs a machine leaning method such as Adaboost or SVM to build a classifier. In detection phase, a sliding window is usually adopted to move across the input image to find the face patches. Among those mentioned methods, a combination Adaboost and Haar-like feature is the most robust, effective, and widely used method in IR domain [7].

However, the performance of using just a single type of feature is limited. In our study, we see that each type of feature has its own advantage and its own range of applications. For examples, Haar-like feature can grasp the contrast characteristics of image areas, while MB-LBP feature is strong in expressing textures in different scales, and HoG feature is good at depicting edges. This encourages us to employ many local features and take all of their advantages simultaneously.

In this paper, we propose a method to mix local features based on AdaBoost in IR domain. We employ the most widely used features for face detection in visible image: Haar-like, MB-LBP, and HoG.

[1] Kyushu University, Fukuoka 819-0395, Japan
[a)] ma@limu.ait.kyushu-u.ac.jp
[b)] trung@limu.ait.kyushu-u.ac.jp

2. Related Works

2.1 Common Features

There are three most commonly used features, Haar-like feature, MB-LBP feature, and HoG feature for face detection with visible cameras. Haar-like feature was first proposed by Viola and Jones [4]. They used Adaboost algorithm to combine many weak classifiers, one with a Haar-like feature, into a strong classifier. Then, multiple strong classifiers are chained together to build a cascade classifier. Their cascade classifier was the first real-time high performance method in face detection in visible image. Later, Reese et al. [7] proved that it was also feasible to use Haar-like feature for face detection in thermal image. LBP feature, short for Local binary patterns, was first described in 1994 by Ojala et al. [8]. MB-LBP, short for Multi-block LBP, was a distinctive rectangle features proposed by Zhang et al. [5]. MB-LBP was designed for face detection using Adaboost in visible image. HoG is short for Histograms of Oriented Gradients, was first proposed by Dalal and Triggs [9] in 2005, and used for human detection in visible image. Zhu et al. [10] proposed a HoG based cascade classifier trained by Adaboost. To the best of our knowledge, there is no application of MB-LBP or HoG features for face detection in IR domain.

2.2 Mixture of Features

In order to enhance the discrimination ability of features, multiple types of features are used together rather than using just a single type. There are three approaches to combine multiple type features: concatenation, co-occurrence, and mixed feature pool.

Concatenation is simply by concatenating individual feature vectors to make a longer feature vector. Wang et al. [11] proposed a method to concatenate HoG histogram and LBP histogram to be HoG-LBP feature. Jiang and Ma [12] created color feature and bar-shape feature, and concatenated them with HoG feature to obtain a feature called HoG III.

Co-occurrence is an extension strategy to increase the variation of the feature pool by using more than one feature simultaneously in one weak classifier. Mita et al. [13] proposed a joint Haar-like feature for face detection, their joint feature was implemented by two or three co-occurred Haar-like features. They showed that their new feature is much better than utilizing single Haar-like feature.

Mixed feature pool is another method to combine more than one type of feature together. This approach mixes two or more type of features into a mixed feature pool for Adaboost selection. Therefore, the built strong classifier may contain different types of features. Xia et al. [14] proposed a mixed feature pool for object tracking application. They mixed two value-type features, Haar-like feature and HoG feature, in their mixed feature pool. We argue that only mixing value-type feature is limited because category-type feature such as MB-LPB cannot be considered. Furthermore, in deciding the threshold for each weak classifier, they just adopted an approximate strategy by averaging the feature responses of positive samples and those of negative samples. It should be better solved by an optimization mechanism. In our work, we use AdaBoost to mix both value-type features, such as Haar-like/HoG feature, and category-type features, such as MB-LBP feature. We also adopt an error minimization strategy to find the optimal thresholds instead of the simple thresholds in Xia el al.'s method.

3. Mixed Features with Adaboost

3.1 Basic Idea

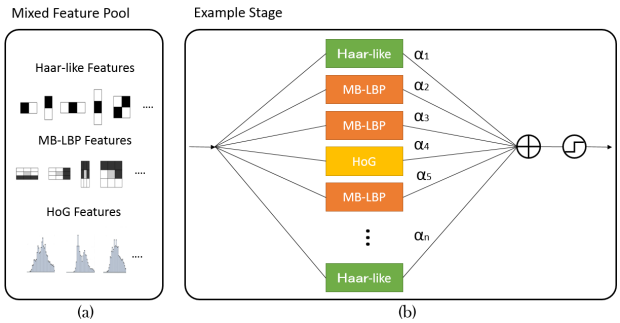

Fig. 1 The idea of using mixed feature pool to build a stage. (a) A mixed feature pool, which contains three different types of features, (b) Example of one stage, each weak classifier is added to the stage by selecting the best feature among all the features in the mixed feature pool, so a stage may consisted of deferent types of features

Different types of features have different representation ability. In order to enhance the discriminative ability of features, we build up a mixed feature pool that contains all Haar-like features, MB-LBP features, and HoG features. Then similar to Viola and Jones approach, we employ Adaboost to train a cascade classifier which is consisted of many stages, each stage contains many weak classifiers. In building a stage, Adaboost algorithm continues to add a new weak classifier to the stage by selecting the optimal feature from the mixed feature pool. Meanwhile, it calculates corresponding weight for the selected feature and updates sample weights until the already built stage meet the requirement of predefined stage minimum detection rate and stage maximum false alarm rate.

In this way, a stage may contain multiple types of features, and each feature is the best one in that iteration of selection among all of the three types of features, we can expect to take advantage of description power from the three types of features.

3.2 Algorithm

The building of a stage by Adaboost is realized by continuing to add a voter which contains a weak classifier and its weight to the current stage. Fig.2 shows the flowchart of our algorithm.

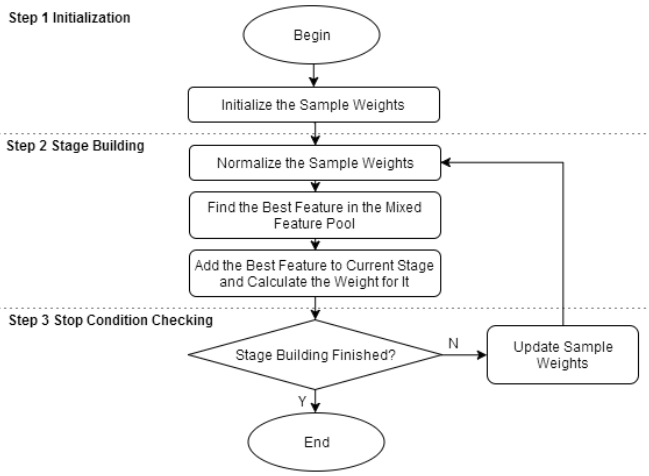

Fig. 2 The algorithm flowchart for training one stage. the stage building is realized by continuing to add the best feature in the feature pool to the stage, and calculating its weight.

The rule to choose the best feature from the feature pool is by minimum error criteria:

$$e_m = \sum_{i=1}^{N} w_{n,i} \left| h_m(x_i) - y_i \right|, \tag{1}$$

where $h_m(x_i) \in \{0, 1\}$ is the prediction function that gives the prediction result of the weak classifier C_m on sample x_i, $w_{n,i}$ is the sample weight with n times of iterations, and $y_i \in \{0, 1\}$ is the true label of the sample. Through the process of minimizing e_m, the algorithm can find the best feature index in the feature pool, also decide the threshold for value-type feature like Haar-like feature or HoG feature, or the look-up-table for category-type feature like MB-LBP feature, since the prediction function for Haar-like feature or HoG feature follows:

$$h_m(x_i) = \begin{cases} 0 & \text{if } d_m R_m(x_i) < d_m T_m \\ 1 & \text{otherwise,} \end{cases} \tag{2}$$

where $R_m(x_i)$ represents the response of the specific feature f_m on samples x_i, T_m is the threshold of the feature, and d_m is a parity which indicates the direction of the inequality sign. On the other hand, the prediction for MB-LBP feature follows:

$$h_m(x_i) = LUT(R_m(x_i)), \tag{3}$$

where $R_m(x_i)$ represents the response of the specific feature f_m on samples x_i. From these two equations we can see the prediction function for value-type feature such as Haar-like/HoG feature and that for category-type feature such as MB-LBP feature are different. For calculating the the error of value-type features, (1) and (2) are combined, and for that of category-type feature, (1) and (3) are combined. The specific feature in the mixed feature pool gives the minimum error is the best feature. The whole algorithm for building a stage is as follow:

- **Input:** Training samples $(x_1, y_1), \ldots, (x_n, y_n)$ where $y_i = 0$ for negative samples and $y_i = 1$ for positive samples.
  User defined training parameter: stage minimum detection rate: DR, and stage maximum false alarm rate: FAR.
- **Output:** The strong classifier, which is one stage in the cascade classifier.
- **Step 1 Initialization:**
  Suppose there are m positive samples and l negative samples, $m + l = n$, initialize the samples weights $w_i = \frac{1}{2m}$ and $\frac{1}{2l}$ for positive and negative samples, respectively.
- **Step 2 Stages Building:**
  (1) Normalize the sample weights so that the sum of all the weights equal to 1:
  $$\tilde{w}_{n,i} \leftarrow \frac{w_{n,i}}{\sum_{j=1}^{n} w_{n,j}}. \tag{4}$$
  (2) Select the best feature f_v with minimum error e_v in the mixed feature pool.
  (3) Add the best weak classifier C_v corresponding to the feature f_v to current stage. The weight of the weak classifier is decided by
  $$\alpha_v = ln\left(\frac{1 - e_v}{e_v}\right). \tag{5}$$
  (4) Update the weight of all training samples for current stage: $w_{n+1,i} \leftarrow \tilde{w}_{n,i}\beta^{1-\lambda}$, where $\lambda = 0$ if the sample x_i is correctly classified by C_v, otherwise $\lambda = 1$, where $\beta = \frac{e_v}{1-e_v}$.
- **Step 3 Stop Condition Checking:**
  (1) Test currently built stage and decide whether it is finished or not. Suppose there are M weak classifiers in current built stage, the voting result of these M weak classifiers on sample x_i is $G(x_i) = \sum_{m=1}^{M} \alpha_m h_m(x_i)$. In order to decide the threshold for current stage, sort the voting result of all samples from small to large, and find the value T where the detection rate equal to DR.
  (2) Use the threshold T to check the false alarm rate of all the training samples, if it is larger than FAR, go to step 2 to continue adding weak classifier to the stage, otherwise, T is the threshold of the stage, and the building of current stage is finished.

4. Experiment

4.1 Dataset

We used PI-450 thermal camera manufactured by Optris with a lens of FOV 62 degree for capturing. We set the camera into raw image mode, by this mode it can record thermal image with temperature ranged from -20°C to 100°C with accuracy of 0.01°C, and mounted the camera on the tripod at the height of 1.6 m.

We have 3 variations in our dataset. First variation is distance, which indicates the location of the person standing in front of the camera. We set 10 different distances from 0.5 m to 5 m with step of 0.5 m. As shown in (a) of Fig.3, each green point represent a capturing location with a different distance. Second variation is appearance, which indicates whether the person wear the glasses or not. We let the person put on glasses to take a set of samples and then take off the glasses to take a set of samples with the same number, so we have two appearances. Third variation is pose, which indicates the head posture of the person, we set 5 directions at each distance as shown in (a) of Fig.3 by blue arrows. For all the directions the person face to, we let the person to look up, look forward and look down. In addition, in real scene, there are more front faces than side faces appear in the images, so we supplemented some more poses of the front face by asking the person to tilt his/her head to left /right shoulder side while looking up, looking forward and looking down. Totally, we have 21 poses with one appearance at each distance.

We employed 20 persons which include 14 males and 6 females as models for capturing the dataset. For each person, firstly we let him/her stand on each distance with the without-glasses appearance, and show all the 21 poses to capture. Secondly, we let the person to repeat the process again with the with-glasses appearance. By this way, we totally have: 20 persons × 10 distances × 2 appearances × 21 poses = 8400 images.

After we obtained the samples, we manually marked the face areas. We define the face areas to be the square with the height from the top of head to the chin, and the (b) of Fig.3 shows the face samples of all the poses marked from one person's images at 1 m without glasses.

4.2 Experiment Settings

In training the cascade classifier, we used the face patches we marked as the positive samples, and the areas in captured images which do not contain a face as the negative samples. We employed leave-one-out cross-validation in our experiment, which left one person's images as testing data and used the other 19 persons' images as training data.

In training phase, we set the stage minimum detection rate 0.995 and stage maximum false alarm rate 0.5, and normalized all the samples size to 24 × 24 for training. We set the number of training samples 7000 for both positive and negative. In detection phase, we used the sliding window based approach. In order to deal with faces in different scales, we first built an image pyramid with multiple layers, we set the scaling factor between pyramid layers to 1.1 and set the minimum face size to 24 × 24, and maximum face size to 150 × 150, because the statistical analysis on our dataset showed that faces in our dataset are within this range.

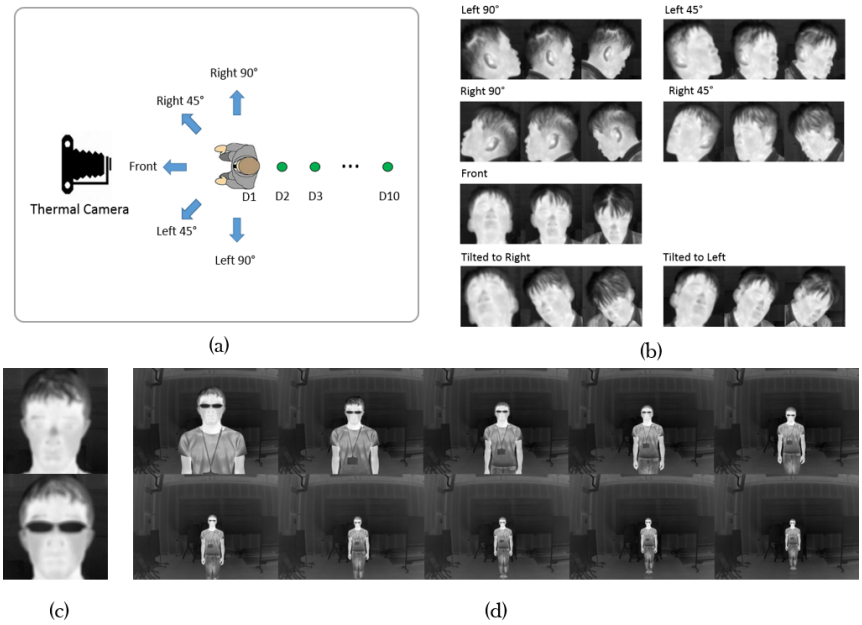

Fig. 3 The setting and variations of our dataset. (a) The setting for capturing the dataset. (b) The samples of the 21 different poses of one person without glasses at 1m. (c) The samples of without-glasses apperance and with-glasses appearance. (d) The samples of one person standing at 10 different distances with glasses.

After face detection was finished in all the layers, the results of all the layers were fused into original scale.

In order to decide whether the detected bounding box is correct detection or false alarm, we used Jaccard Index [15] as the judging criteria:

$$Jaccard\ Index\,(B_1, B_2) = \frac{Area\,(B_1 \cap B_2)}{Area\,(B_1 \cup B_2)}, \qquad (6)$$

where the B_1 represents the marked bounding box of the ground truth face area, and B_2 represents the detected bounding box. We deem the face successfully detected when $Jaccard\ Index\,(B_1, B_2)$ is larger than 0.5.

4.3 Results and Discussion

We used recall and precision for evaluating the cascade classifiers. We calculated the recall and precision of cascade classifier with different features and stages for all divisions. In this way we obtained 20 sets of curves. We combine to show the 20 sets of curves in Fig. (4). In the figure, one curve represents the performance of one feature category, one marker on the curve indicates the average recall and average precision of a specific number of stages. we used the number of stages from 12 to 21.

From the figure we can see, first, with the same precision, our mixed feature with some number of stages gives the best recall. With respect to the number of stages, for any number of stages on the curves of individual features, we can find a point with some number of stages on the curve of mixed features that has higher recall and precision, which means mixed features give the best performance. Similarly, the second best one is the Haar-like feature. The performance of MB-LBP feature and HoG feature is far from mixed features and Haar-like feature.

Second, we can see a trade-off between recall and precision from all the curves. If we increase the number of stages, the precision is increasing, but the recall is decreasing.

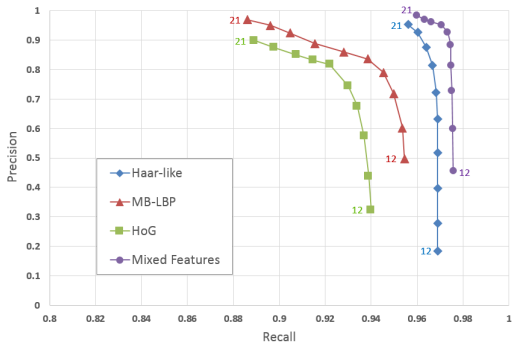

Fig. 4 Combined results of leave-one-out cross-validation.

5. Conclusion and Future Works

In this paper, we propose an algorithm to combine different types of features. We created mixed feature pool that contains three most commonly used features: Haar-like feature, MB-LBP feature, and HoG feature, together, for thermal face detection, and employed Adaboost to build a cascade classifier by using the mixed feature pool. We captured a thermal image dataset of 8400 images, then we used leave-one-out cross-validation to compare the performance of our mixed features with those of the regular features. The experiment results show that the performance of our mixed features can significantly dominate that of the regular features. For regular features, Haar-like features gives the best performance.

In the future we intent to increase the number of the samples, and add more sample variations to our dataset.

References

[1] Zhang, Y., Lu, Y., Nagahara, H., and Taniguchi, R. I.: Anonymous camera for privacy protection, *ICPR*, pp. 4170-4175 (2014).

[2] Wong, W. K., Hui, J. H., Desa, J. B. M., Ishak, N. I. N. B., Sulaiman, A. B., and Nor, Y. B. M.: Face detection in thermal imaging using head curve geometry, *Image and Signal Processing (CISP)*, pp. 881-884 (2012).
[3] Wang, S., Liu, Z., Shen, P., and Ji, Q.: Eye localization from thermal infrared images, *Pattern Recognition*, 46(10), 2613-2621 (2013).
[4] Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features, *CVPR*, Vol. 1, pp. I-511 (2001).
[5] Zhang, L., Chu, R., Xiang, S., Liao, S., and Li, S. Z.: Face detection based on multi block LBP representation, *International Conference on Biometrics*, pp. 11-18 (2007).
[6] Paisitkriangkrai, S., Shen, C., and Zhang, J.: Face detection with effective feature extraction, *Asian Conference on Computer Vision*, pp. 460-470 (2010).
[7] Reese, K., Zheng, Y., and Elmaghraby, A.: A comparison of face detection algorithms in visible and thermal spectrums, *Int'l Conf. on Advances in Computer Science and Application*, (2012).
[8] Ojala, T., Pietikainen, M., and Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on pattern analysis and machine intelligence*, 24(7), 971-987 (2002).
[9] Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection, *CVPR*, Vol. 1, pp. 886-893 (2005).
[10] Zhu, Q., Yeh, M. C., Cheng, K. T., and Avidan, S.: Fast human detection using a cascade of histograms of oriented gradients, *CVPR*, Vol. 2, pp. 1491-1498 (2006).
[11] Wang, X., Han, T. X., and Yan, S.: An HOG-LBP human detector with partial occlusion handling, *ICCV*, pp. 32-39 (2009).
[12] Jiang, Y., Ma, J.: Combination features and models for human detection, *CVPR*, pp. 240-248 (2015).
[13] Mita, T., Kaneko, T., and Hori, O.: Joint haar-like features for face detection, *ICCV*, Vol. 2, pp. 1619-1626 (2005).
[14] Xia, C., Sun, S. F., Chen, P., Luo, H., and Dong, F. M.: Haar-like and HOG fusion based object tracking, *Pacific Rim Conference on Multimedia*, pp. 173-182 (2014).
[15] Karlinsky, L., Dinerstein, M., Levi, D., and Ullman, S.: Combined model for detecting, localizing, interpreting and recognizing faces, *Workshop on Faces in'Real-Life'Images: Detection, Alignment, and Recognition*, (2008).