Adaptive Contour Features in Oriented Granular Space for Human Detection and Segmentation

Wei Gao, Haizhou Ai
Computer Science and Technology Department
Tsinghua University, Beijing, P.R. China
ahz@mail.tsinghua.edu.cn

Shihong Lao
Core Technology Center
Omron Corporation, Kyoto, Japan
lao@ari.ncl.omron.co.jp

Abstract

In this paper, a novel feature named Adaptive Contour Feature (ACF) is proposed for human detection and segmentation. This feature consists of a chain of a number of granules in Oriented Granular Space (OGS) that is learnt via the AdaBoost algorithm. Three operations are defined on the OGS to mine object contour feature and feature co-occurrences automatically. A heuristic learning algorithm is proposed to generate an ACF that at the same time define a weak classifier for human detection or segmentation. Experiments on two open datasets show that the ACF outperform several well-known existing features due to its stronger discriminative power rooted in the nature of its flexibility and adaptability to describe an object contour element.

1. Introduction

Human detection has attracted lots of attention in computer vision fields in the past decades due to its wide spread of potential applications including automated visual surveillance, smart human computer interaction, driver assistant systems, etc. Nowadays, although face detection or at least frontal face detection is a well accepted solved problem in academic society, human detection remains a big challenge due to a wide viewpoint change of highly articulated body postures, varying illumination conditions and occlusions. The framework of human detection is more or less similar to face detection. However not as face detection which has been approached mainly on texture features due to its more consistent appearance pattern, human detection usually relies on gradient contour or silhouette features in order to across additional changes due to clothing, articulation and occlusion. In literature review, we find that the kind of features selected in successive learning algorithms plays a crucial rule in performance of the developed detectors [1, 21, 15, 19]. Therefore in this paper, we dedicate to develop a more discriminative and robust feature which can mine object contour and feature co-occurrences adaptively for object detection.

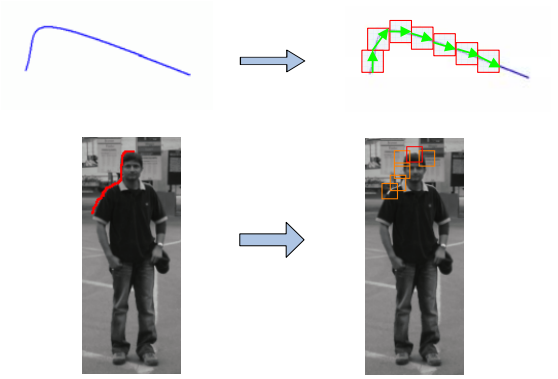


Figure 1. Describe Contour with Adaptive Contour Feature

In nature, object detection is an asymmetric classification problem since we can just describe what a positive sample looks like while we cannot tell what a negative sample should be. Enlighten by this idea, we target at developing features that can strongly describe an object. Since gradient or contour features have been proved with great advantage in human detection [1, 21, 15, 19], the mission changes to find a flexible and robust way to describe object contour elements. Inspired by this idea, we propose a novel feature named Adaptive Contour Feature (ACF) as shown in Figure 1. An ACF is defined as a chain of square patches called granules in Oriented Granular Space (OGS) to describe a curve. The feature value of ACF is designed to measure the similarity between the contour described by ACF and the actual one. Some simple operations in the OGS and a heuristic learning algorithm are proposed to mine object contour features and feature co-occurrences automatically.

The contribution of this paper has three aspects: first, a novel feature called Adaptive Contour Feature (ACF) is proposed which has great discriminative power and can mine object contour features adaptively and feature co-

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

occurrences automatically; second, a heuristic learning algorithm is developed to learn ACF and its corresponding weak classifier for AdaBoost at the same time; third, a coarse human segmentation is presented with ACF learned in detection.

The structure of this paper is as follows. In Section 2 we review the related work on the human detection. In Section 3 we give the detail of the ACF and some parameter refinement issue for the feature. A coarse segmentation is presented by boosting on the learned ACF in Section 4. In Section 5 we present some experiments to show the ACF's privileged performance in human detection and segmentation. Finally, we conclude in Section 6.

2. Related Work

The research works on human or pedestrian detection before 2005 could be found in a survey [14] and also in an experimental study [13]. We will mainly focus on more recent research works that were roughly divided into three categories as follows.

The first category is building generative models via shape matching. Gavrila [4] detected humans using a hierarchical template matching to find all candidates and a verification RBF network classifier was applied to cut down false alarms. Lin *et al*. [11] proposed a hierarchical part-template matching method that could handle partial occlusions and articulations flexibly.

The second category is learning from low-level local image features. Volia *et al*. [20] applied Haar-like features from both appearance and motion in AdaBoost learning. Dalal and Triggs [1] used Histogram of Oriented Gradient (HOG) features in training Support Vector Machines (SVMs). Later, Zhu *et al*. [25] used HOG features in AdaBoost learning that significantly improved efficiency. Wu and Nevatia [21] introduced local edge features called edgelets and trained both full body detector and part detectors to handle occlusions. Sabzmeydani and Mori [15] presented a set of mid-level features called shapelet in a two fold AdaBoost learning. Tuzel *et al*. [19] used covariant matrices as image descriptors in LogitBoost learning on Riemannian Manifolds. And finally, Wu and Nevatia [24] integrated heterogeneous features in AdaBoost learning to achieve higher performance.

Instead of using low-lever image features directly, the third category uses the Implicit Shape Model (ISM) to build a codebook for human detection. Leibe *et al*. [8] learned visual words descriptor for humans and built a generative model for detection and segmentation, where a top-down verification step by Chamfer is applied to reduce false alarms. Shotton *et al*. [18] first built a class dictionary of spatially localized contour fragments from the input segmentation masks and then learned the detector by boosting on these contours fragments. Ferrai *et al*. [2, 3] used adjacent contour segments mined from edges to build a codebook for object detection.

For multi-pose and multi-view human detection, Hou *et al*. [5] divided all variations into three views to learn a detector using Vector Boosting. Since sometimes human viewpoint is difficult to category even manually, Wu and Nevatia [22] proposed a cluster boosted tree classifier which categories samples into groups automatically by feature clustering, while Shan *et al*. [17] clusted humans into categorization by learning discriminative exemplars.

As for human segmentation, Wu and Nevatia [23] learned human segmentation in the same frame work as human detection by boosting on local edgelet feature. Lin *et al*. [11, 10] used hierarchical part-template matching to segment human.

Our work is most related to [1, 21, 2]. However, our ACF can describe an object contour more accurately than HOG [1]. Compared with predefined pixel level edgelet [21], ACF is learned from the training samples, which is more invariant with pose variation. And different from [2] in which contour fragments are learned from segmented input masks, ACF is mined out in learning discriminative features for detection and is object independent.

3. Adaptive Contour Features

HOG features [1] and edgelet features [21] have been proved very effective in human detection, however there are some limitations. HOG features just compute the gradient distribution in a rectangle which cannot describe the actual object shape well. And an edgelet feature is defined on pixel level which suffers to change with object deformation and articulation. Especially, these features are all predefined ones that neither take into account the prior knowledge of particular object category nor consider its feature co-occurrences. To overcome these limitations, the proposed ACF in this paper is expected to have the following merits: first, it can capture contour elements of an object category well which means learning what its shape looks like; second, it is robust with reasonable object deformation as HOG; third, it can mine its shape feature co-occurrences automatically, for example, human's shoulder and leg co-occurrence.

3.1. ACF in Oriented Granular Space

The Granular Space as defined in [7] is a special scale space of the original image in which each layer (bitmap) corresponds to the granules (a square window patch) of a certain scale. Originally a granule is defined as a triplet $G = (x, y, s)$, where (x, y) is the position, s is the size or scale (in practice we use 2×2, 4×4, 8×8). In this paper, we extended it to a quaternion group $G = (x, y, s, o)$, where an additional parameter o is the orientation of the granule as

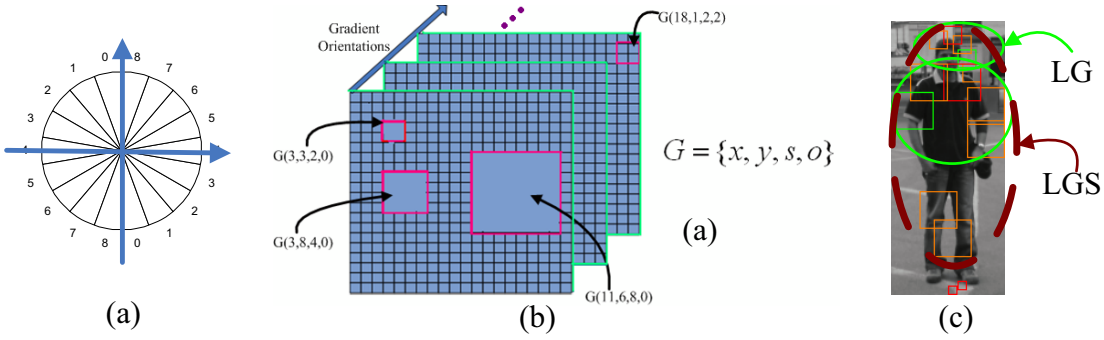


Figure 2. (a) orientation bins (b) Oriented Granular Space. Each layer represents granules in one orientation. (c) definition of LG and LGS.

shown in Figure 2(b), which we called the Oriented Granular Space (OGS).

As illustrated in Figure 1, the essential idea of ACF is to represent object contour elements as a chain of a number of granules. As HOG [1], given an input image I, the gradient magnitude M and edge orientation O over the image are calculated of which the orientation is quantized into $N_b = 9$ bins $[k\frac{\pi}{N_b}, (k+1)\frac{\pi}{N_b}]$, where $k = 0, ..., N_b - 1$(shown in Figure 2(a)). $M(u, v)$ and $O(u, v)$ are gradient magnitude and orientation of image I respectively at position (u, v). Two kinds of feature values of a granule are calculated (as equations 1 and 2), which are accumulated edge strength at its orientation: $E(G(x, y, s, o))$ and its relative strength over all orientations: $\overline{E}(G(x, y, s, o))$ in its corresponding square window patch.

$$E(G(x, y, s, o)) = \sum_{(u,v)\in G} M(u, v)\delta(o, O(u, v)) \quad (1)$$

$$\overline{E}(G(x, y, s, o)) = \frac{E(G(x, y, s, o))}{\sum_{(u,v)\in G} M(u, v)} \quad (2)$$

$$\delta(i, j) = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \quad (3)$$

The two feature values together represent the edge response in the specific orientation of the granule G. The larger the response, the higher the probability of a true edge with the orientation of the granule is. And different size of the granules can describe the edge in different scale level. Small granules describe fine edges while large granules describe coarse edges.

With granules and their feature values defined, we introduce the Linked Granules (LG) and Linked Granules Set (LGS) shown in Figure 2(c) as following:

1. **LG** is defined as a chain of a number of granules to describe a continuous contour as equation 4.

$$LG = \{G_i | i = 1...k\} \quad (4)$$

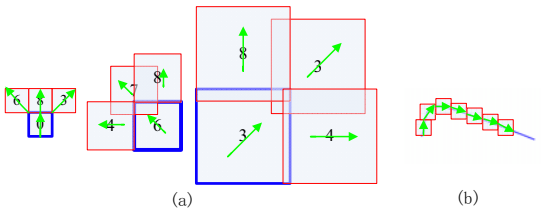


Figure 3. (a) Grow orientation of different granule. The blue granule is current one and the red granules are the candidates to grow. Green arrows are granules' orientation. (b) A example of Linked Granules generated by grow operation.

The linkage method of the granules is learned in Section 3.2. The strength and the relative strength of a LG are defined as the sum over all its granules:

$$E(LG) = \sum_{G_i \in LG} E(G_i) \quad (5)$$

$$\overline{E}(LG) = \sum_{G_i \in LG} \overline{E}(G_i) \quad (6)$$

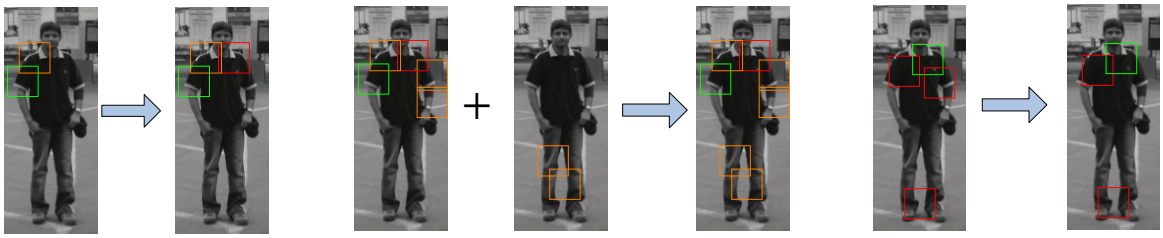
2. **LGS** is defined as combination of several LGs to describe co-occurrence features. The feature values of LGS are defined in equations 7 and 8.

$$E(LGS) = \sum_{LG_i \in LGS} E(LG_i) \quad (7)$$

$$\overline{E}(LGS) = \sum_{LG_i \in LGS} \overline{E}(LG_i) \quad (8)$$

One thing to mention, in implementation those feature values of OGS can be computed hierarchically in preprocessing that guarantees a very high efficiency.

Finally we define an ACF as a LGS. An ACF is learned in OGS with a heuristic algorithm (see section 3.2) under three operations (shown in Figure 4): ***Grow***, ***Combine*** and ***Cut***.



(a) Grow Operation (b) Combine Operation (c) Cut Operation

Figure 4. Operations defined on Oriented Granular Space (OGS).

1. **Grow Operation:** Grow an existing LG by adding a candidate granule of the same scale on its head or tail with a constraint on contour continuity:

$$LG_{new} = LG \cup \{G\} \tag{9}$$

As shown in Figure 3(a), a LG is grown from its head or tail (in blue) as current granule, in which the added granule (in red) should be its neighbor and be consistent in orientation. By consistent we mean in the implementation that both the orientation difference of the two linked granules and the orientation difference between the line linking their centers and the candidate's orientation must be at most $\pm 45°$(as shown in Figure 3).

2. **Combine Operation:** Combine several LGs or equivalently grow an existing LGS by adding a candidate LG to represent co-occurrence features:

$$LGS_{new} = LGS \cup \{LG\} \tag{10}$$

Mita *et al*. [12] have proved that some feature co-occurrences of object can improve the discriminative power greatly. As shown in Figure 4(b) co-occurrence of human shoulder and leg provides a stronger evidence of human appearance. In our framework, feature co-occurrences are also automatically found in the feature learning procedure.

3. **Cut Opeartion:** Cut off a granule from a head or tail of an existing LG:

$$LG_{new} = LG - \{G\} \tag{11}$$

In summary, ***Grow*** operation searches for object contour elements; ***Combine*** operation mines feature co-occurrences and ***Cut*** operation refines for better features. Since edge features are computed in granule level of different scale, ACF is robust with object appearance changes under reasonable deformation and articulation as HOG feature.

3.2. Heuristic Learning Algorithm for ACF

We adopt the cascade detector framework of Viola and Jones [20] and used Real AdaBoost [16] in training a nested cascade detector [6] for human detection. However, instead of selecting features from a predefined feature pool, we construct ACF features and their corresponding weak classifiers and then select the most discriminative one heuristically in each boosting round as in Figure 5 over training set.

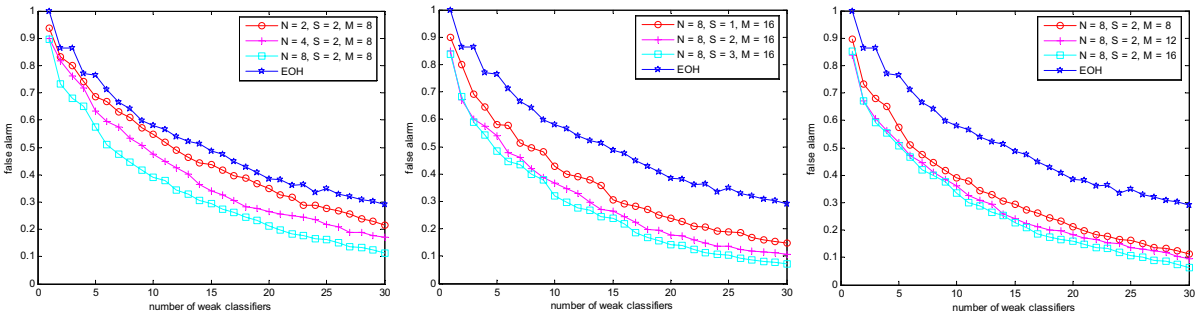
- Init LGS feature pool as Open List Feature Pool $(OLFP)$
- For search iteration $SI < MAX\_ITERATION$
  - Select the most discriminative $SEED$ LGS features from OLFP, add them to Close List Feature Pool $(CLFP)$ and remove them from $OLFP$
  - For each seed feature j selected
    - Find the most discriminative features $Grow_j$, Com_j and Cut_j with operations ***Grow***, ***Combine*** and ***Cut***
  - Add $Grow_j$, Com_j and Cut_j to $OLFP$
  - $SI++$
- Construct Heuristic Feature Pool (HFP) from $OLFP$ and $CLFP$
- Train the best weak classifier from HFP

Figure 5. Heuristic algorithm to construct ACF

The open list is initialized with all single granules in the OGS. For each loop, some best seeds are selected from the open list and expanded by the ***Grow***, ***Combine*** and ***Cut*** operations defined in Section 3.1. Finally, a best ACF is selected from the constructed feature pool and its corresponding weak classifier is fused to the strong classifier.

3.3. ACF Parameters

In practice, there are three main parameters in ACF learning: the number of granules in a LG allowed (N), that is the length of contour element; the number of LGs in a LGS allowed (S), that is, the degree of feature co-occurrences allowed; and the total number of granules in a LGS allowed (M), which affects the feature complex-



(a) Length of LG (b) Segments of LGS (c) Total Granules of LGS

Figure 6. Adaptive Contour Feature Parameters and Comparison with EOH.

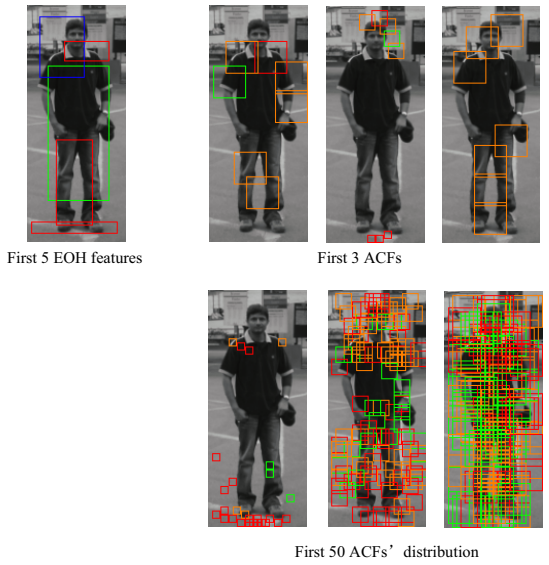


Figure 7. Selected EOH features and ACFs

ity and computation efficiency. In order to analyze the discriminative power of ACF with different parameter settings, we compared with the AdaBoosted Edge Orientation Histogram (EOH) features [9] that are in fact corresponding to single granule features of this paper but with rectangle window patches learned by Real AdaBoost, which is in fact an enhanced HOG feature [5].

We collect 3,700 human samples and 3,700 non-human ones of the size 24×58. Both EOH features and ACFs are trained on this set with Real AdaBoost. We search 30 rounds for each boosting round ($MAX\_ITERATION = 30$), and use 10 features as seeds ($SEED = 10$) for each searching round. As in Figure 6, it can be seen that ACFs are much more discriminative than EOH features and converge to lower false alarm rate more quickly.

In Figure 6(a), the discriminablity improves as the number of granules in a LG allowed (N) increases from 2 to 8, which means longer contour is better. While in Figure 6(b), the degree of feature co-occurrences allowed (S) could have respective impact on the performance, which means more co-occurrence features are preferred. And in Figure 6(c), we can see that the more the total number of granules in a LGS allowed (M) the better the performance. However, M has a direct impact on computation expense. And given a certain M, the choice of N and S must be compromised according to the experiments in Figure 6(a)(b). In our implementation, for efficiency we set $N = 8$, $S = 2$, and $M = 8$. Some EOH and ACF features learned in the early boosting rounds are shown in Figure 7. It can be seen that ACFs describe human contour elements and feature co-occurrences well.

4. Simultaneous Detection and Segmentation

ACF is constructed mainly by ***Grow*** and ***Combine*** operations, where ***Grow*** operation mines object contour elements and ***Combine*** operation mines feature co-occurrences. Inspired by ACF's advantage in describing object contour elements, we apply ACFs in human segmentation in the same way as Wu and Nevatia [23] to show their power on characterizing the object contours.

For input samples of the size $w \times h$, a cascade classifier is trained for detection, and a number of $w \times h$ classifiers are trained simultaneously for per-pixel segmentation. The samples for detection are denoted as $S^d = \{(x_i, y_i)\}$, where $y_i = \pm 1$, and samples for segmentation are $S^s = \{x_i, +1, m_i\}$, where m_i is segmentation mask of the same dimension with x_i. The foreground pixels are treated as positive samples in segmentation while the background pixels as negative. In each boosting round, an ACF is learned for detection as described in Section 3.2. Then the ACF is applied to the pixels in the ACF's effective field for segmentation. The effective field of an ACF's is defined as the pixels located in the granules of the ACF. The segmenta-

- Given an intial set for detection $S^d = \{(x_i, y_i)\}$,where $y_i = \pm 1$, and a sample set for segmentation $S^s = \{x_i, +1, m_i\}$,where m_i is mask, and a human free negative sample set.
- Set the cascade layer L; the maximum weak classifier T; the detection false alarm F; the cascade classifier H_c; the segmentation classifier H_s
- Initialize the segmentation sample weight D^s, where $D^s(u)$ is a pixel weight at position u; the cascade layer $l = 0$
- While $l < L$
  - Initialize the detection sample weight D^d; set layer false alarm $f = 1.0$, layer boosting round $t = 0$, layer detection classifier h_d
  - While $t < T$ and $f > F$
    - Construct and select an ACF as described in Section 3.2
    - Add selected ACF to h_d, and update D^d
    - Add ACF to H_s in its effective field and update D^s
    - Update threshold and recompute false rate f
    - $t++$
  - Recollect negative samples for detection from human free images
  - $l++$
- Output the cascade classifier H_c for detection and the strong classifier H_s for segmentation

Figure 8. Simultaneous detection and segmentation with ACF.

tion classifier is trained together with the cascade structure detector as shown in Figure 8.

As in the detection procedure, there may be many detection responses for one human. The segmentation classifier is applied to each of these responses. The final segmentation result is achieved using a simple voting method.

5. Experiment Results

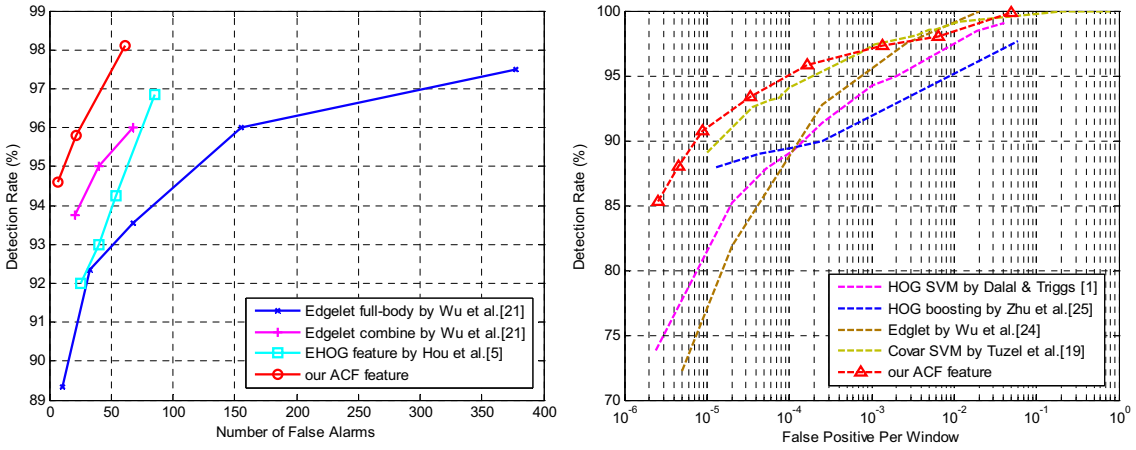
Detection experiments are carried out on two different datasets: one is USC Pedestrian Set A [21] which contains 205 real-life photos and 313 different humans with front/rear and has no heavy inter-object occlusion, another one is INRIA dataset [1] which is more challenging with pose variation, inter-object occlusion and complex background.

Since USC Pedestrian Set A has no training set, we collect 3,700 front/rear human samples (7,400 after left-right reflections). The samples are normalized to 24×58 and negative samples are collected from 9,000 human-free images. The parameter setting is the same as described in Section 3 which is a tradeoff between feature's discriminability and its computation efficiency. We compare our ACF with edgelet feature (Wu and Nevatia [21]) and EHOG feature (Hou *et al*. [5]) in Figure 10(a). It can be seen in Figure 10(a) that ACF obviously outperforms the other two features.

INRIA dataset is a complete set with both train and test samples. It consists of 1,239 pedestrian images (2,478 with their left-right reflections) and 1,218 person-free images for training. In the test set there are 566 pedestrian examples (1,132 with their left-right reflections) and 453 person-free images. All the human samples are cut from images and scaled to 24×58. We also use the Cascade structure and collect negative samples from person-free images with a bootstrap procedure. Detection rate under different false alarm level is achieved by tuning layers of the Cascade. Unlike the ROC curve in Figure 10(a), miss-rate/FPPW (False Positive Per Window) curves is adopted as shown in Figure 10(b). Some results on INRIA set and our own images are shown in Figure 11.

Figure 9. Samples for segmentation train.

In order to train human segmentation, we labeled 620 (1,240 after left-right reflection) humans of the size 24×58 (some samples are shown in Figure 9). As described in Section 4, the segmentation classifiers are trained directly with the ACF learned for detection. Some results of segmenta-



(a) Result on USC Set A

(b) Result on INRIA

Figure 10. (a) ROC curve comparison on USC Set A. (b) detection-rate/FPPW curve on INRIA Set .

tion after detection are shown in Figure 12.

The cascade detector can scan about 680,000 windows per second. And in average for detection and segmentation it takes totally about $120ms$ on a 400×300 image with a popular PC.

6. Conclusion

A novel feature called Adaptive Contour Feature (ACF) is proposed for object detection and segmenation. An Oriented Granular Space (OGS) is first constructed from a gradient image, and three simple operations are defined to generate ACF features from a single granule. We propose a heuristic algorithm to construct ACFs with high discriminative power as weak classifiers. ACFs can describe object contour elements adaptively in different levels and search for feature co-occurrences automatically. It is also robust with reasonable object deformation and articulation as HOG feature. Experiments on two open human detection datasets: USC Set A and INRIA Set, show the advantage of ACFs in object detection. To demonstrate the contour description ability of ACFs, a coarse human segmentation is also done with the ACFs learned in detection to show its description power. Although ACF is proposed for human detection and segmentation, it can also be used in other object detection and segmentation problems.

Acknowledgements: This research was supported in part by National Basic Research Program of China (2006CB303102), Beijing Educational Committee Program (YB20081000303), and it was also supported by a grant from Omron Corporation.

References

[1] N. Dalal and B. Trigges. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 1, 2, 3, 6

[2] V. Ferrai, T. Tuytelaars, and L. Gool. Object detection by contour segment networks. In *ECCV*, 2006. 2

[3] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Grops of ajacent contour segments for object detection. *Transactions on Pattern Analysis and Machine Intelligence*, 30(1):36–51, 2008. 2

[4] D. Gavrila. Pedestrian detection from a moving vehicle. In *ECCV*, 2000. 2

[5] C. Hou, H. Ai, and S. Lao. Multiview pedestrian detection based on vector boosting. In *ACCV*, 2007. 2, 5, 6

[6] C. Huang, H. Ai, B. Wu, and S. Lao. Boosting nested cascade detector for multi-view face detection. In *ICPR*, 2004. 4

[7] C. Huang, H. Ai, L. Yuan, and S. Lao. Learning sparse features in granular space for multi-view face detection. In *FG*, 2006. 2

[8] B. Leibe, E. Seemann, and B. Schiele. Pedestrain detection in crowed scenes. In *CVPR*, 2005. 2

[9] K. Levi and Y. Weiss. Learning object detection from a small number of examples: the importance of good features. In *CVPR*, 2004. 5

[10] Z. Lin and L. Davis. A pose-invariant descriptor for human detection and segmentation. In *ECCV*, 2008. 2

[11] Z. Lin, L. Davis, D. Doermann, and D. DeMenthon. Hierarchical part-template matching for human detection and segmentation. In *ICCV*, 2007. 2

[12] T. Mita, T. Kaneko, B. Stenger, and O. Hori. Discriminative feature co-occurrence selection for object detection. *Transactions on Pattern Analysis and Machine Intelligence*, 30(7):1257–1269, 2008. 4

[13] S. Munder and D. Gavrila. An experimental study on pedestrain classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1863–1868, 2006. 2

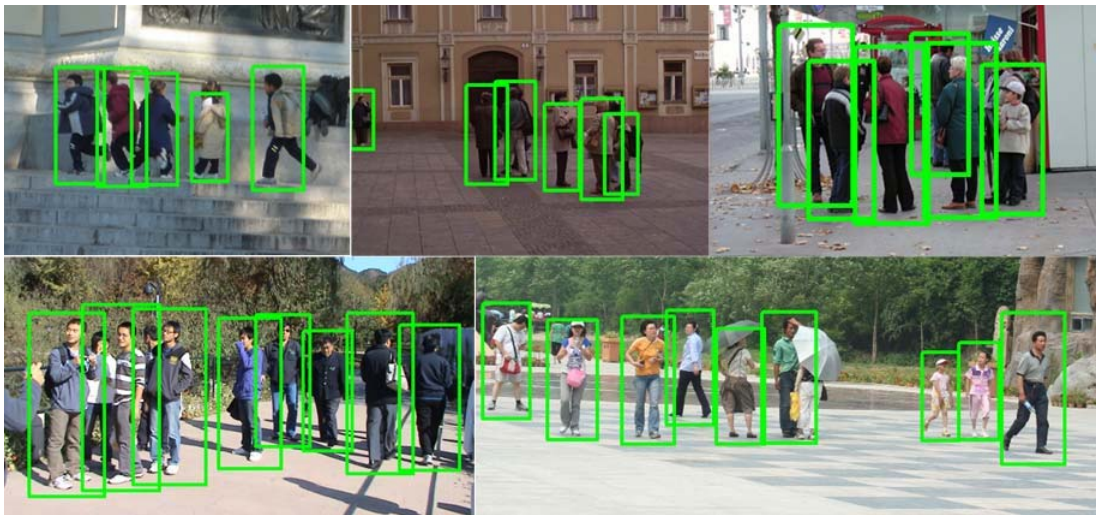


Figure 11. Detection results.

Figure 12. Segmentation results.

[14] N. Ogal. A survey of techniques for human detection from video. In *University of Maryland, Technical report*, 2005. 2

[15] P. Sabzmeydani and G. Mori. Detecting pedestrians by learning shapelet features. In *CVPR*, 2007. 1, 2

[16] R. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, 37. 4

[17] Y. Shan, F. Han, H. Sawhney, and R. Kumar. Learning exemplar-based categorization for the detection of multi-view multi-pose objects. In *CVPR*, 2006. 2

[18] J. Shotton, A. Black, and R. Cipolla. Contour-based learning for object detection. In *ICCV*, 2005. 2

[19] O. Tuzel, F. Porikli, and P. Meer. Human detection via classification on riemannian manifold. In *CVPR*, 2007. 1, 2

[20] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using pattern of motion and appearance. In *ICCV*, 2003. 2, 4

[21] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *Proc. 10th Int. Conf. Computer Vision*, 2005. 1, 2, 6

[22] B. Wu and R. Nevatia. Cluster boosted tree classifier for multi-view, multi-pose object detection. In *ICCV*, 2007. 2

[23] B. Wu and R. Nevatia. Simultaneous objection detection and segmentation by boosting local shape feature based classifier. In *CVPR*, 2007. 2, 5

[24] B. Wu and R. Nevatia. Optimizing discrimination-efficientcy tradeoff in integrating heterogeneous local features for object detection. In *CVPR*, 2008. 2

[25] Q. Zhu, S. Avidan, M. Yeh, and K. Cheng. Fast human detection using a cascade for histograms of oriented gradients. In *CVPR*, 2006. 2