CO-OCCURRENCE FLOW FOR PEDESTRIAN DETECTION

Atsuto Maki[1] *Akihito Seki*[2] *Tomoki Watanabe*[2] *Roberto Cipolla*[1,3]

[1] Toshiba Research Europe, Cambridge Research Laboratory, U.K.
[2] Research and Development Center, Toshiba Corporation, Japan
[3] Department of Engineering, University of Cambridge, U.K.

ABSTRACT

The last few years have seen considerable progress in pedestrian detection. Recent work has established a combination of oriented gradients and optic flow as effective features although the detection rates are still unsatisfactory for practical use. This paper introduces a new type of motion feature, the *co-occurrence flow* (CoF). The advance is to capture relative movements of different parts of the entire body, unlike existing motion features which extract internal motion in a local fashion. Through evaluations on the TUD-Brussels pedestrian dataset, we show that our motion feature based on co-occurrence flow contributes to boost the performance of existing methods.

Index Terms – HOG, motion feature, flow, pedestrian

1. INTRODUCTION

Pedestrian detection is a highly active research area of computer vision, involving various techniques to improve feature design, classification as well as non-maximal suppression. The applications range from surveillance to image indexing and notably automotive safety [14, 8, 11, 6, 9] which this paper is also concerned with. A recent benchmark [5], using a large dataset recorded from a moving vehicle, provides an overview of state-of-the-art performance of a number of detection algorithms [2, 16, 4, 7, 12, 20]. It reported that histograms of gradients (HOG) [2] remains competitive, while the detection rates of the best methods still require large improvements for practical applications. One of the common challenges in most of these vision systems is to deal with varying appearance of pedestrians under different walking phases as well as viewing directions by using static image features. An exception is [16], which first employed motion features, although a static camera was assumed.

Motion is an important cue, especially for a monocular system, which enables us to see what is not noticeable in a single image. Although there have been few works which incorporate motion in pedestrian detection [16, 3, 21, 13], it was recently shown that additional use of motion features to HOG can enhance the performance for on-board sequences, in particular for pedestrians with side views which are of high importance in automotive safety applications [21]. The added motion feature, originally introduced in [3] as histograms of flow (HOF) feature, was computed for example by applying wavelet-like operators on a 3×3 local cell grid of HOG.

In this paper, motivated by the previous work [21] which justified the use of motion, we introduce a novel motion feature, the *co-occurrence flow* (CoF). The idea of co-occurrence flow is to capture possible coherence in movements of different body parts into a motion feature. It is also inspired by the co-occurrence histograms of oriented gradient [18], which are obtained by pairwise voting of edge orientations. In our case, in order to encode the unique motion of walking into our feature, we design the CoF feature through pairwise comparisons of histograms of optic flow for the entire body, i.e. across exhaustive combinations of cells defined typically by a 4×8 grid of squares, forming a rectangular region. See Fig. 1 for the sketch of the CoF feature.

We compute probabilities of being pedestrian for candidate regions in terms of a combination of CoF feature and a multi-level version of the HOG descriptor [2]. As the classifier we choose to employ the linear SVM and HIKSVM, support vector machines with histogram intersection kernel [12], because of the performance and the popularity.

In the remainder of the paper, Section 2 describes CoF, the new motion feature, together with our implementation of HOG. Section 3 explains the setting of our pedestrian detection and shows the performance of our detector in experiments in comparison with that of HOF feature combined with HOG. Section 5 is the conclusion.

2. CO-OCCURRENCE FLOW

Our motion feature is motivated by the fact that strong correlations exist in the movements of different body parts when a pedestrian is walking. They include correlations between the motion of two legs, those between two parts of an arm, or those between a leg and an arm. The correlations provide useful cues to identify the walking motion unique to pedestrians [13]. It is at least the case for human vision, as shown for example by a well known experiment of Johansson [10].

In our system, we aim to capture this discriminative power in optic flow correlations which we call *co-occurrence flow*

978-1-4577-1302-6/11/$26.00 ©2011 IEEE

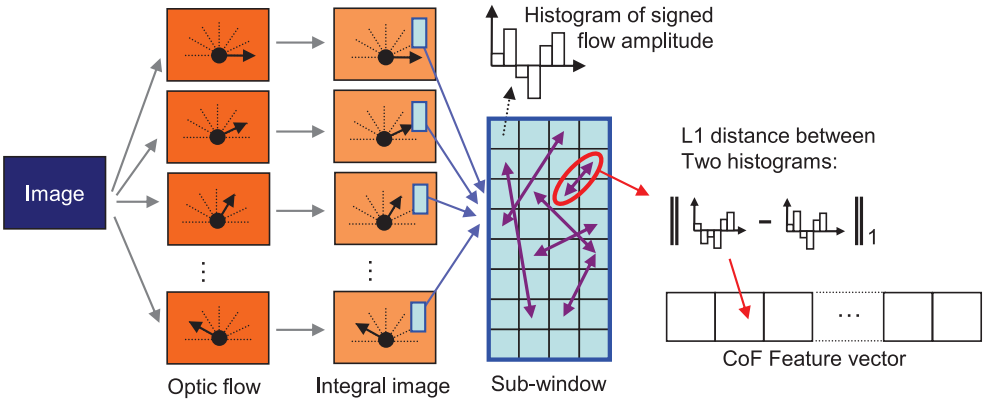

Fig. 1. An example of our motion feature, *co-occurrence flow* (CoF) feature vector, computed for a sub-window consisting of 4×8 cells. We first compute the oriented edge energy responses and store them in discretised channels. (In this case we consider six orientations and therefore use six integral images accordingly.) We then compute a local histogram of signed flow amplitude for each cell, and make pairwise comparisons of histograms across exhaustive combinations. See text in Section 2 for details.

(CoF). Namely, when certain flow is observed on one body part, simultaneous flows on other parts are likely to be related to it in specific ways, either coherent or dissimilar with some characteristic differences depending on the offsets of the body parts. We encode this notion in the CoF feature (see Fig. 1). We design it to be suitable for window search so that it can be utilised jointly with a HOG feature, which is also the case with our detector. This section introduces the design details of the CoF feature, preceded by the description of our HOG implementation.

2.1. Multiscale HOG

Our approach to extract HOG features starts with computations of the oriented edge energy responses by convolving the input image, $\mathcal{I}$, with oriented odd Gabor filters in $d\ (=8)$ different orientations, which is reported to improve results [12]. As we use integral images [1] to efficiently compute our features in windows of various size, this filtering is first performed for the entire image, $\mathcal{I}$, rather than separately for each overlapping detection window. We denote the outputs of Gabor filtering in the j-th direction as $G(j)$ for $j = 1, ..., d$.

Given a candidate rectangular region, R, of an arbitrary size and with the aspect ratio of 1:2 for finding pedestrians, we define cells, subregions of R, in each of which we compute elements of HOG feature. The cells are gridwise generated in a multi-level fashion so that we have $2^l \times 2 \cdot 2^l (l = 0, ..., l_{max})$ cells in each level. We choose $l_{max} = 3$ as a reasonable number for the finest level. To give a rough idea, this indicates that each cell in the bottom level consists of 8×8 pixels for an R with size 64×128 pixels.

For a cell at a level l, which we refer to as $w_l(m, n)$ for $m = 1, ..., 2^l, n = 1, ..., 2^{l+1}$, we construct a set of feature elements, $\mathbf{f}_l(m, n) \in \mathbf{R}^d$, by computing the sum of the outputs of Gabor filtering, $\{G(j)\}$, at each orientation channel within the corresponding subregion. That is,

$$\mathbf{f}_l(m,n) = \{e_l(m,n;j)\}\ ,\ j = 1, ..., d \quad (1)$$

$$e_l(m,n;j) = \left| \int_{w_l(m,n)} G(u,v;j) du dv \right| \quad (2)$$

where (u, v) are local coordinates in R. We then normalise $\mathbf{f}_l(m, n)$ by using filter outputs over all directions. Thus, our normalised feature, $\tilde{\mathbf{f}}_l(m, n)$, consists of entries $\tilde{e}_l(m, n; j), j = 1, ..., d$:

$$\tilde{e}_l(m,n;j) = e_l(m,n;j) \ / \ \frac{1}{d} \sum_{j=1}^{d} e_l(m,n;j) \quad (3)$$

Now, we incorporate outputs at coarser scales which are generally known to be useful (see e.g. [15]) and form an N_G-dimensional HOG descriptor, $\mathbf{v}_G$, by concatenating the features of different levels by $\mathbf{v}_G = [\tilde{\mathbf{f}}_3\ \tilde{\mathbf{f}}_2\ \tilde{\mathbf{f}}_1\ \tilde{\mathbf{f}}_0]$ where $N_G = d \sum_{l=0}^{l_{max}} 2^l \cdot 2^{l+1}$. $N_G = 1360$ when $d = 8$ and $l_{max} = 3$.

2.2. CoF feature

Co-occurrence flow (CoF), uses pairwise comparisons between local histograms of optic flow as its building blocks. Given a rectangular region, R, we generate $a\ (= m \times n)$ cells in the same way as the second finest level in computing HOG; $l = 2$ so that $a = 4 \times 8$ (see Fig. 1). In each cell, $w_k(m, n)$ for $m = 1, ..., 2^k, n = 1, ..., 2^{k+1}, k = 2$, we compute a local histogram of optic flow, $H(m, n)$, by voting the pixels according to the orientations of flows into $b\ (= 6)$ bins while using their flow magnitudes as weighting factors.

We use the technique of [19] for computing a regularised flow field for the entire image, $\mathcal{I}$. For the sake of computational efficiency, the flow field is stored in separate channels $F(i)$ for $i = 1, ..., b$ (one per discretised orientation). Each $F(i)$ is represented using integral images. Thus, each bin of the local histogram, $H(m, n)$, can be effectively produced by accessing the subregion of $\{F(i)\}$ which corresponds to the

cell of interest, $w_k(m,n)$. That is, the i-th element of the histogram is computed as

$$h(m,n;i) \quad = \quad \int_{w_k(m,n)} F(u,v;i)dudv \ , \ i = 1,...,b. \quad (4)$$

Given an on-board camera, the computed flow field is naturally influenced by possible camera motion. Our strategy to cope with the influence of camera motion is to subtract the dominant background flow from the original flow before generating the local histogram. This is in contrast to the previous approach of using derivatives of differential flow such as in Internal Motion Histogram descriptors [3, 21]. In practice, we compute the dominant flow by averaging the flow globally observed in R. We can compute the i-th element of the histogram considering this subtraction simply by

$$h'(m,n;i) = h(m,n;i) - \int_R F(u,v;i)dudv \ . \quad (5)$$

We then make the pairwise comparison of $H'(m,n)$ for all possible $N_F (= {}_aC_2)$ combinations inside R. We have $N_F = 496$ for $a = 4 \times 8$. Using the L_1 norm as the measure[1], each comparison outputs a scalar, $S_{AB} = |H'(A) - H'(B)|_1$ where A and B are indices to arbitrary cells, and thereby we obtain an N_F-dimensional vector, $\mathbf{v}_F = \{S_{AB}\}$, which encodes our CoF feature.

3. EXPERIMENTS

Our pedestrian detection is based on a window search. As explained earlier, we take computational efficiency into consideration in several aspects of our detector; we compute CoF and HOG features using integral images in order to reduce the cost of window search. By facilitating the access to subwindows at arbitrary positions in varying scales, our CoF and HOG features computation is done in a GPU implementation. We search for pedestrians by extracting bounding-boxes at every 4 pixels along both horizontal and vertical directions across the input image. We examine 17 different scales ranging from 0.4 to 2.0, corresponding to 50 and 256 pixels of the height. Each detection is counted as correct if it overlaps with an annotation by more than 50% using the intersection-over-union measure.

Once we run a search and detect numerous regions of interest (ROIs) that are classified as pedestrians, we apply a non maximal suppression (NMS) algorithm to merge the detections. The approach we take is to smooth the 3D map of output detection scores across 2D coordinates and scale, and then to find their local peaks which we select as positive outputs. The dimensions of the feature vectors, $\mathbf{v}_G$ and $\mathbf{v}_F$, are, $N_G = 1360$ and $N_F = 496$, respectively, given $a = 4 \times 8$. Our feature, $\mathbf{v} = [\mathbf{v}_G \mathbf{v}_F]$, therefore has the total of 1856 dimensions.

[1]We found the performance better than the case of using histogram intersections.

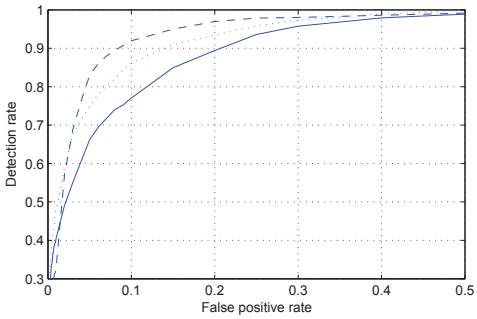

Fig. 2. The ROC curves generated for TUD-Brussels dataset using: (i) CoF+HOG features (in dashed), (ii) multiscale HOG features (in solid), and (iii) HOF+HOG features (in dotted). Overall, CoF+HOG features perform the best among the three.

3.1. Training scheme

We use the TUD-MotionPairs database [21] for the purpose of training as the images are provided as sets of consecutive frames so that training based on motion features is possible. The positive dataset taken in a busy pedestrian zone consists of 1092 pairs of images, containing 1776 annotations of pedestrians. The negative dataset consists of 192 images containing no pedestrians. We take a few approaches to increase the number of training samples; we extract several samples per annotation by wobbling the region which we access for computing the features while considering mirroring as well. We then train the classifier: support vector machines with linear or histogram intersection kernel.

After training with those initial samples and therewith running the classification using the same training data set, we acquire numerous examples of false positive (hard negative) as well as true negative. We then proceed the bootstrapping training by using those additional samples.

3.2. Evaluations

For evaluations we use the TUD-Brussels database [21] that is recorded from a car driving at varying speed. It consists of 508 pairs of images containing 1326 annotated pedestrians. In order to evaluate the entire scheme, we study the performance in plots in terms of recall and precision by a simulation using positive (annotated) pedestrian regions and randomly selected negative rectangular regions (note that the resulting recall rates appear relatively higher in terms of ROC curves than the cases of using actual detections). We then compare the results of using (i) CoF+HOG features with those from (ii) HOG features only. We also compare the CoF+HOG features with (iii) a HOF+HOG features, a state-of-the-art motion feature `IMHd2` [17], which is a recent modification of HOF. The dimension of this motion feature alone is 2520 per rectangular region, which is reduced from the original HOF feature [3].

In Fig. 2, we show the ROC curves obtained with those three types of features. Linear support vector machines are

Fig. 3. Examples of pedestrian detections by HOG+CoF features. Detected bounding boxes are shown in green. The second picture shows colour coded optic flow map for the frame shown in the left.

used for this analysis. The performance increase by our detector using CoF+HOG features is significant over the case of using only HOG features. The difference in the obtained recall is by 14.9% at a false positive rate of 90%. CoF+HOG features also outperforms HOF+HOG features by 5.8% at the same error rate although the recall rate deteriorates at false positive rates lower than 2%. It should be noted that CoF features achieve the performance with the dimension that is five times smaller (half in total including HOG) than that of the alternative HOF feature.

Fig. 3 shows examples of pedestrian detection by HOG + CoF features with HIKSVM (detected bounding boxes overlaid). The optic flow map is also provided for the leftmost example. It appears evident in the flow map that motion can serve as a strong cue, however it is also observed that issues may arise due to occlusions; in this example one of the targets, the third from the right, was not detected when HOG+HOF features were used instead although it is with our HOG+CoF features as can be seen. Overall these examples illustrate that CoF features efficiently capture the motion of pedestrians.

4. CONCLUSION

We have introduced a new motion feature, the *co-occurrence flow* (CoF), which improves the performance of pedestrian detection in combination with the HOG descriptor. The idea is to globally capture relative movements of different body parts. We evaluated the performance of CoF feature through experiments, and showed that our detector using the CoF feature boosted the performance by combining it with a standard HOG feature even though a limited amount of data has been analysed so far.

Future work will be first directed to a more thorough evaluations with a larger dataset. It will be also interesting to investigate how the two complementary features can be unified in other efficient ways. Finally, the concept of co-occurrence flow can have broader applications to recognition of motion such as those of human actions in video processing.

Acknowledgement

The authors are very grateful to Frank Perbet for helpful discussions.

5. REFERENCES

[1] F. Crow. Summed-area tables for texture mapping. *SIGGRAPH*, pages 207–212, 1984.

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR (1)*, pages 886–893, 2005.

[3] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV (2)*, pages 428–441, 2006.

[4] P. Dollár, Z. Tu, H. Tao, and S. Belongie. Feature mining for image classification. In *CVPR*, pages 1–8, 2007.

[5] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: A benchmark. In *CVPR*, pages 304–311, 2009.

[6] M. Enzweiler and D. M. Gavrila. Monocular pedestrian detection: Survey and experiments. *IEEE-PAMI*, 31(12):2179–2195, 2009.

[7] P. F. Felzenszwalb, D. A. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, pages 1–8, 2008.

[8] D. M. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *IJCV*, 73(1):41–59, 2007.

[9] H. Hattori, A. Seki, M. Nishiyama, and T. Watanabe. Stereo-based pedestrian detection using multiple patterns. In *BMVC*, 2009.

[10] G. Johansson. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 14(2):201–211, 1973.

[11] B. Leibe, N. Cornelis, K. Cornelis, and L. J. V. Gool. Dynamic 3d scene analysis from a moving vehicle. In *CVPR*, pages 1–8, 2007.

[12] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *CVPR*, pages 1–8, 2008.

[13] F. Perbet, A. Maki, and B. Stenger. Correlated probabilistic trajectories for pedestrian motion detection. In *ICCV*, pages 1647–1654, 2009.

[14] A. Shashua, Y. Gdalyahu, and G. Hayon. Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. In *IV: IEEE Intelligent Vehicles Symposium*, pages 1–6, 2004.

[15] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *CVPR*, pages 1–8, 2008.

[16] P. A. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *ICCV*, pages 734–741, 2003.

[17] S. Walk, N. Majer, K. Schindler, and B. Schiele. New features and insights for pedestrian detection. In *CVPR*, pages 1030–1037, 2010.

[18] T. Watanabe, S. Ito, and K. Yokoi. Co-occurrence histograms of oriented gradients for pedestrian detection. In *PSIVT*, pages 37–47, 2009.

[19] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, and H. Bischof. Anisotropic Huber-L1 optical flow. In *BMVC*, 2009.

[20] C. Wojek and B. Schiele. A performance evaluation of single and multi-feature people detection. In *DAGM-Symposium*, pages 82–91, 2008.

[21] C. Wojek, S. Walk, and B. Schiele. Multi-cue onboard pedestrian detection. In *CVPR*, pages 794–801, 2009.