3D Extended Histogram of Oriented Gradients (3DHOG) for Classification of Road Users in Urban Scenes

Norbert Buch
norbert.buch@kingston.ac.uk
James Orwell
j.orwell@kingston.ac.uk
Sergio A. Velastin
sergio.velastin@kingston.ac.uk

Digital Imaging Research Centre
Kingston University
Kingston upon Thames, UK

Abstract

This paper proposes and demonstrates a novel method for the detection and classification of individual vehicles and pedestrians in urban scenes. In this scenario, shadows, lights and various occlusions compromise the accuracy of foreground segmentation and hence there are challenges with conventional silhouette-based methods. 2D features derived from histograms of oriented gradients (HOG) have been shown to be effective for detecting pedestrians and other objects. However, the appearance of vehicles varies substantially with the viewing angle and local features may be often occluded. In this paper, a novel method is proposed that overcomes limitations in the use of 2D HOG. Full 3D models are used for the object categories to be detected and the feature patches are defined over these models. A calibrated camera allows an affine transform of the observation into a normalised representation from which '3DHOG' features are defined. A variable set of interest points is used in the detection and classification processes, depending on which points in the 3D model are visible. Experiments on real CCTV data of urban scenes demonstrate the proposed method. The 3DHOG feature is compared with features based on FFT and simple histograms. A baseline method using overlap between wire-frame models and motion silhouettes is also included. The results demonstrate that the proposed method achieves comparable performance. In particular, an advantage of the proposed method is that it is more robust than motion silhouettes which are often compromised in real data by variable lighting, camera quality and occlusions from other objects.

1. Introduction

In recent years, there has been an increased scope for automatic analysis of urban traffic activity. This is due in part to the additional numbers of cameras and other sensors, the enhanced infrastructure and consequent accessibility and also advances in analytical techniques to detect traffic violations (illegal turns, one way streets, *etc*) and to identify road users. Using general purpose surveillance cameras, the classification of vehicles is a demanding challenge (see Figure 1). Compared to most examples in the image retrieval field, the quality of surveillance data is generally poor and the range of operational conditions (night-time, inclement and changeable weather that affects the auto-iris) require robust techniques which need to be immune to errors in obtaining road users' silhouettes. In consultation with a government transport department, we use five generic categories for our classifier: Bus/Lorry; Van; Car/Taxi; Motorbike/Bicycle and Pedestrian.

Figure 1 Example views from the i-LIDS dataset with detected and classified pedestrians and vehicles. The image on the right illustrates the 3D models used.

Our contribution is three-fold. Firstly, 3D spatial models are introduced to define the location of interest points from which local features are extracted. The local features are constructed out of histograms of oriented gradients (HOG). The combination of 3D interest points and HOG is hence introduced as the novel 3DHOG feature. Performance is evaluated, comparing 3DHOG with FFT and histogram-based local features. The second contribution is a training and classification framework based on the 3DHOG feature which allows classification using a variable number of interest points (previous approaches required a fixed number of interest points). This approach works independently of motion silhouettes and can be applied to stationary objects, still images or moving cameras and is therefore in principle less likely to be affected by motion segmentation issues. Our third contribution is an extensive evaluation of the proposed method on real video benchmarking data (i-LIDS from UK Home Office) publicly available from [2].

The remainder of the paper is organised as follows: The next section discusses related work. Section 3 introduces the feature extraction process that is used in section 4 for training. The classification framework is introduced in section 5 with performance evaluation in section 6. The paper concludes with section 7.

2. Related work

The process of classifying images or objects in images can be generally categorised either as top-down (usually visual surveillance) or bottom-up (usually object recognition) approaches. For top down, the whole context is analysed simultaneously or used to verify a hypothesis during searching. Motion silhouettes are generated from background modelling and classification is performed based on motion silhouette measurement features [20,24,18]. This approach is vulnerable to inaccurate foreground segmentation, which is inherent to urban environments due to low camera angles, occlusions, *etc*. Effort has been directed to accurate foreground segmentation by various shadow removal techniques or the instantaneous background, as in [8]. The above 2D approaches can be extended to 3D for vehicle detection and classification as in [24,18] and Buch et *al.* [3,4]. The motion silhouette outline is used for classification in [18,3] and for vehicle detection of a single size in [22]. Wire frames are matched to images in [26].

In contrast to the above, bottom up approaches are usually targeted at object categorisation and classification of still images. An extensive range of local features have been proposed: SIFT [16], SURF [1], GLOH [19], boundary fragment model BFM [21], HOG [6,7] with an overview in [19]. Together with those features, the use of spatial constraint is desirable to improve performance. A simple 'bag of words' approach is often not sufficient as it does not localise objects. In [10], a fixed spatial model for feature

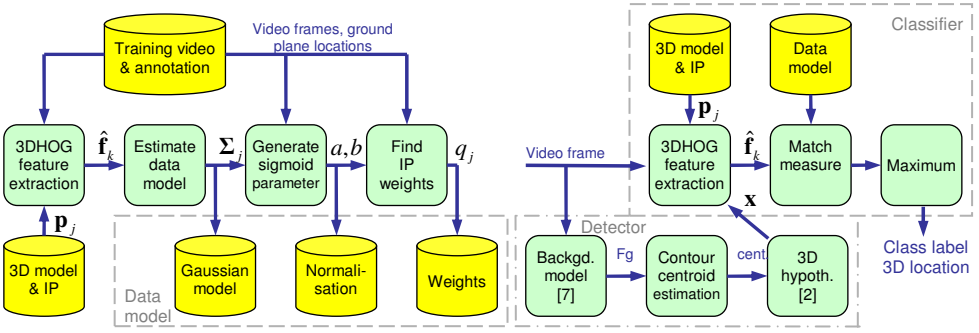


Figure 2 Block diagram of the framework for training and classification

grouping in highway scenes is used. A constellation model is used in [17] for vehicle classification. Most spatial constraints can be expressed with the *k*-fans introduced in the seminal paper of Crandall et *al.* [5]. The 'implicit shape model' is used in Leibe et *al.* [14] for pedestrian detection. Extensions of this early work in [12, 13] show the object recognition community moving towards surveillance applications [15, 23]. However, the obtained performance figures are not yet good enough for practical real world applications.

Top-down and bottom-up approaches are combined by Dalal and Triggs [6], using local features with 2D fixed spatial constraints. This is used for pedestrian detection, vehicle classification into 2 classes [17] and action recognition (temporal extension) [11].

2.1. 3DHOG detector and classifier

Our approach takes the good results from 3D models into account [24,18,3] and defines the local features and the spatial relationship between them in 3D world space. The top down solution from histogram of oriented gradients (HOG) using a 2D search window [6] is generalised to 3D by 'wrapping' the camera image around the models like in [25]. Using calibrated cameras, obtained in a relatively straightforward way given a plan map of the scene, the scale is determined directly, in contrast to the multiple scale search in [6]. By introducing a framework that deals with variable numbers of visible interest points, we can use a single model to detect objects from any angle (Figure 2). Our algorithm detects rigid vehicles and pedestrians in the same way and does not require special cases. The algorithm uses texture to generate local features only and does not rely on potentially noisy motion information. This implies that the method is applicable in cases where reliable motion information is not available, *e.g.* stationary objects, single frames and moving cameras.

3. Local features in 3D

First we define the position of a set of interest points located on the faces of 3D models (Figure 1 similar to [3]) of the objects to be classified. Then for a candidate object (either during training or when classifying) we obtain image patches for interest points that are sufficiently visible. Finally, we calculate feature vectors from those patches. The method, described next, is applied to all models and to keep the expressions succinct there is no model index subscript. An interest point $\mathbf{p} = \left[x, y, z, e_x, e_y, e_z \right]$ is determined by its 3D location $\left[x, y, z \right]$ and orientation $\mathbf{e} = \left[e_x, e_y, e_z \right]$. A set of interest points $\mathbf{P} = \{\mathbf{p}_j\}$ on a face is defined on a regular grid with face density d_f around an origin $\mathbf{p}_0$ (centre of the face) and with direction normal to the face

Figure 3 Extraction pipeline: 3D model with interest points $\mathbf{P}$ followed by input image I, extracted image patches $\mathbf{I}_p$ and feature vectors $\hat{\mathbf{f}}_k$. The radius of cones indicates the weight q_j of interest points $\mathbf{p}_j$. If a cone is missing from the grid, it was either not visible during training or gave poor performance and was rejected accordingly.

$$\mathbf{p}_j = \mathrm{grid}\left(\mathbf{p}_0, d_f\right). \tag{1}$$

To ensure good coverage for small faces of *e.g.* pedestrians, while also limiting the total number of interest points for large faces of *e.g.* buses, the face density d_f is adjusted according to face size s_f (maximum extent) relative to a reference size s_0 and a growth parameter γ:

$$d_f = \frac{d_0}{(1-\gamma)+\gamma^{s_f/s_0}} \tag{2}$$

For our experiments $s_0 = 4\mathrm{m}$, $\gamma = 0.35$ and $d_0 = 4$ were used, which trades off oversampling against the use of too large patches, which would lead to global rather than local features and hence to loss of discriminating power.

3.1. Extracting image patches

The extraction process automatically resolves the scale and perspective distortion of the observation and presents a constant size image to a classifier. The locations of interest points are used to extract visible image patches I_{pk} from an image I at a given object model location $\mathbf{x} = [x, y, z, r]$ with orientation r. Let $\mathbf{v}_j$ be the viewing direction of interest point $\mathbf{p}_j$. The visible set of interest points $\mathbf{P}_v = \{\mathbf{p}_k\} \subseteq \mathbf{P}$ is determined by the visibility threshold $\tau_v = 0.65$ to ensure minimum visibility:

$$\mathbf{P}_v = \left\{\mathbf{p}_j \,\middle|\, \langle \mathbf{e}_j, \mathbf{v}_j \rangle > \tau_v \right\}. \tag{3}$$

A square image patch I_{pk} is defined for interest points $\mathbf{p}_k$ with pixel size $l_p = \delta \cdot \rho$ using constant 3D world resolution ρ in pixels per metre and width δ in metres allowing some overlap of patches. An affine transformation with bilinear interpolation is used to map pixels of the input image I to images I_{pk} producing the set of visible image patches $\mathbf{I}_p$. The cardinality of the set $\mathbf{I}_p$ is variable depending on the viewing direction of the model. The process can be viewed as one of wrapping the camera image around the model resulting in invariant representations for any 3D location and viewpoint. See Figure 3 for an example of the processing pipeline. Histogram stretching is applied to individual images I_{pk} to achieve additional illumination independence.

3.2. Generating patch features

The image patches $\mathbf{I}_p$ extracted as explained in the previous section are used to generate normalised feature vectors $\hat{\mathbf{f}}_k$. The length of those vectors depends on the algorithm used, but the training and classification framework is independent of that length. Vectors $\mathbf{f}_k$ provided by any one of the available algorithms (HOG, FFT or Histogram) are normalised for better performance, according to [6]:

$$\hat{\mathbf{f}}_k = \frac{\mathbf{f}_k}{\|\mathbf{f}_k\|_2} \tag{4}$$

3D Histogram of Oriented Gradients (3DHOG)

The generation of the feature vectors $\mathbf{f}_k$ for image patches $\mathbf{I}_p$ is performed in the same way that Dalal and Triggs [6] generate the vectors for single cells. First, a Sobel kernel $[-1,0,1]$ is used to compute the gradient image for all three colour channels independently. The angles are calculated in the range $[0,2\pi]$ as this is recommended for rigid objects like vehicles. A single histogram is generated for every image patch with η bins. The highest gradient magnitude of the three channels is used for the histogram. We use the visible part of 3D models to extract patches, which can be seen as '3D windows' generalising the concept of planar 2D windows in the seminal paper [6]. This adds complexity for combining the variable number of feature vectors $\mathbf{f}_k$, which is efficiently dealt with by a new framework in section 5.

FFT feature

Fast Fourier transform (FFT) features $\mathbf{f}_k$ are calculated from the spectrum of image patches $\mathbf{I}_p$. The DC component is removed to eliminate the influence of illumination. The remaining magnitude spectrum is used to fill a two dimensional histogram with number of angle bins η and number of frequency bins ν. This is similar to using banks of Gabor filters and accumulating the response into a feature vector.

Histogram feature

The grey level histogram is one of the simplest image features that can be used in the classification framework proposed here and thus it is used to compare with the performance of the 3DHOG features. The number of bins is defined by η.

4. Training

The classification framework uses training for every available 3D model. The spatial extent of interest points $\mathbf{P}$ is predefined to generate a data driven model for individual interest points, as outlined below. An overview of the training process is given in Figure 2.

4.1. Data driven model for interest points

Interest point appearances are modelled with single Gaussian distributions. For the estimation of the mean $\boldsymbol{\mu}_j$ and covariance matrix $\boldsymbol{\Sigma}_j$ of every interest point $\mathbf{p}_j$, a training set is used. The training set comprises frame images with a set of model locations $\mathbf{L} = \{\mathbf{x}_N\}$. Those positions $\mathbf{x}_N$ were generated with the baseline algorithm in [3] and manually refined. The approach described in section 3 is used to extract feature vector

samples for interest points in a training frame. Typically $N = 5$ to 30 sample vectors per interest point $\mathbf{p}_j$ are accumulated into sample set $\mathbf{S}_j = \{\hat{\mathbf{f}}_{Nj}\}$ from the training videos. The covariance matrices $\mathbf{\Sigma}_j$ are estimated from sample sets $\mathbf{S}_j$ as diagonal matrices due to the typical cardinality. The Mahalanobis distance measure d_k is used to compare newly seen visible feature vectors $\hat{\mathbf{f}}_k$ with the model.

4.2. Weights for refinement

After estimating the Gaussian models for every interest point, the detection and localisation performance of every individual point can be improved considerably. We have to deal with individual interest points, because SVM classification as in [6] is not possible due to the variable number of interest points in our case. As responses of different points can vary, the average response for different models can be inconsistent. The three refinement steps outlined below overcome those limitations by normalising these responses and automatically determining higher weights for good interest points.

Distance surface

In the first step, a distance surface is calculated for models placed onto the training positions $\mathbf{x}_N$. A regular grid of positions $\mathbf{g}_{MN}$ is generated for every position $\mathbf{x}_N$ in training set $\mathbf{L}$. The size of the grid is set to 4m with 9 steps. This corresponds to a shift between grid points of approximately half an image patch and a total displacement of twice the patch size in every direction. Based on those dimensions, the validation procedure can assess the location sensitivity of interest point data models. The distance between the interest point's models $\mu_j, \mathbf{\Sigma}_j$ and the extracted feature vectors $\hat{\mathbf{f}}_j$ at model positions $\mathbf{g}_{MN}$ gives a distance surface $\mathbf{D}_{MNj}$ for every interest point $\mathbf{p}_j$ at every training position $\mathbf{x}_N$. A mean distance surface $\bar{\mathbf{D}}_{Mj}$ over all training samples N is defined by

$$\bar{\mathbf{D}}_{Mj} = \frac{\sum_N \mathbf{D}_{MNj}}{N}. \tag{5}$$

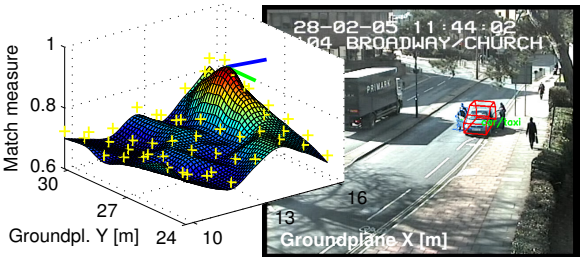
Transfer function

A logistic sigmoid function is estimated to transform a given Mahalanobis distance measure d_k of visible feature vector $\hat{\mathbf{f}}_k$ into a match measure m_k in the interval $[0,1]$. The function uses parameters a and b

$$m_k = \frac{1}{1+e^{a(b-d_k)}}, \tag{6}$$

which are derived from the shape of the distance surface. The middle of the sigmoid function is aligned with the centre score of the distance surface resulting in $b = \bar{\mathbf{D}}_{M/2,j}$. A line is defined between this centre score and the mean of all scores which has gradient g. Using the first derivative of m_k, parameter $a = 4g$ which makes the gradients equal (proof is in the supplementary material). This provides normalised responses of points. The nature of equation (6) limits the influence of large distances (outliers). Any visible subset of interest points will provide the same match measure for models after this normalisation. The match measure response at training positions is given as

$$\mathbf{M}_{Mj} = \frac{1}{1+e^{a\left(b-\bar{\mathbf{D}}_{Mj}\right)}}. \tag{7}$$



symbol	*value*	*unit*
γ	0.35	
d_o	4	
τ_o	0.65	
ρ	32	P/m
δ	1	m
η	10	
ν	4	
τ_M	0.38	

Figure 4 Left: Example of car detection with occlusion of pedestrians showing match measure surface with a good peak. Right: Parameters used during evaluation.

Interest point weight

Relative weights are given to interest points in order to favour those with good localisation performance and reject those with bad performance. For classification, the weight is used to calculate a total weighted average match measure m over visible interest points $\mathbf{p}_k$. A histogram $H_{hj} = \underset{M}{\text{hist}}\left(\mathbf{M}_{Mj}\right)$ of the match surface $\mathbf{M}_{Mj}$ is calculated where every bin h corresponds to a ring of the surface. Low variance of the match measure H_{hj} inside such a ring is a good indicator for consistent and symmetric localisation performance. The interest point weight q_j is calculated from a weighted average of those variances using the element count C_{hj} of histogram bins:

$$q_j = 1 - \sum_h \frac{\text{var}(H_{hj})}{C_{hj}}. \tag{8}$$

To complete the training, the best 80% of interest points are used for the classifier with q_j used as weight. Refer to Figure 3 for a car example with marked up interest points as cones. During classification, variable numbers of visible interest points $\mathbf{P}_v = \{\mathbf{p}_k\}$ contribute to the total match measure

$$m = \frac{\sum_k m_k q_k}{\sum_k q_k}. \tag{9}$$

5. 3D classification framework

The classification framework used here is based on the framework described by Buch et *al.* [3]. Background estimation with a Gaussian mixture model [9] and shadow removal is used to generate motion silhouettes. For each silhouette, a grid of 3D object hypotheses is generated from the centroid and scored by the classifier using equation (9). Please refer to Figure 2 for a block diagram. The silhouettes are often noisy due to the challenging video data in urban environments with changing lighting conditions and low camera angle, but are a good indicator for the existence of a vehicle. A particular problem is the auto iris function of cameras, which adjusts when large white vehicles pass the camera producing large foreground areas during this period of time (Example in results of Figure 6).

The classifier sweeps through models and locations by scoring hypotheses based on only appearance and texture to find the highest match measure above the detection threshold τ_M. In the process, the 3DHOG framework is used to extract visible image

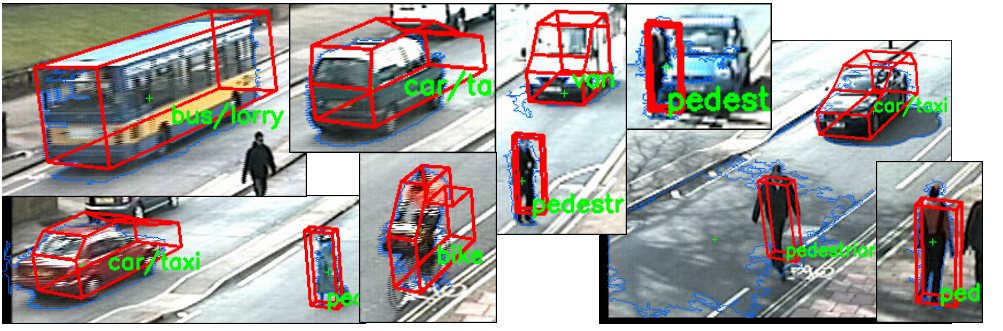


Figure 5 True positive examples for vehicles and pedestrians using 3DHOG.

patches and features for every hypothesis as described in section 3. To handle variable visibility and occlusion, an average match measure per hypothesis is calculated according to equation (9) producing a match surface shown in Figure 4 left. To limit the search space, orientations of vehicles are assumed to align with the road direction, which is realistic for many road videos. The classification is performed on a per frame basis without tracking or temporal refinement.

6. Evaluation

Evaluation was performed on realistic (operational quality) videos for traffic surveillance. All three algorithms are compared with state of the art classifiers. Figure 4 right provides a parameter list for the tests. We use scenario 1 of the Parked Car data set, which is part of the i-LIDS data sets [2] licensed by the UK Home Office for image research institutions and manufacturers. Each dataset comprises 24 hours of video sequences under a range of realistic conditions. They are used by the UK government to benchmark video analysis products and therefore are ideal for evaluating and comparing algorithms in the computer vision community and there is a gradual increase in take-up. Approximately one hour of video for sunny, overcast and changing conditions was selected. The auto iris function of the camera causes image changes for large vehicles. In addition, the overcast videos contain saturated areas in the middle and far end of the view. The car is the most common vehicle type in the dataset. Some illustrative examples are shown in Figure 1 and Figures 5 to 6.

The evaluation is based on an extended confusion matrix including FP (false positives) and FN (false negatives) for detector and classifier (Table 1). Precision P and recall R can be calculated from the confusion matrix [2, 3]. The last row of the matrix is the normalised bounding box overlap between ground truth and detection. A high overlap indicates good localisation performance. The bounding box for our detection is calculated from the wire frame outline of the best fitting model.

Out of the three features in section 3, the best performing algorithm is 3DHOG (Table 1) with a total recall of 81.1% at precision of 82% and classification accuracy of 92.1%. This compares well to recall of 88.2% at precision 89% for the motion silhouette baseline from [3] run on the same data set, but 3DHOG should be better dealing with noise and particularly occlusion. The bounding box overlap of both algorithms is identical 0.68. The system using FFT features showed lower performance (Recall 64.9% at precision 56.5%) but is still able to perform unbiased classification. The detection and classification

a b c d

Figure 6 Four examples generated with 3DHOG. a) Missed car due to low contrast of the vehicle bonnet and roof. b) Misclassified SUV as van due to similar size and appearance. c,d) Correct detections, the blue outline indicates the foreground mask. Due to the auto-iris function of the camera, the mask is too large for the large white vehicles. The 3DHOG classifier can correctly locate and classify the vehicles because it does not rely on the foreground mask.

ground truth / *detection*	bike	car/taxi	van	bus/lorry	FP
bike	1.00	.00	.00	.00	.44
car/taxi	.00	.83	.21	.03	.10
van	.00	.00	.67	.33	.08
bus/lorry	.00	.02	.02	.65	.00
FN	.00	.14	.10	.00	.00
count	27	361	48	40	
overlap	.70	.66	.73	.76	

a

ground truth / *detection*	bike	car/taxi	van	bus/lorry	FP
bike	1.00	.17	.34	.42	5.70
car/taxi	.00	.75	.39	.05	.32
van	.00	.02	.28	.07	.10
bus/lorry	.00	.00	.00	.47	.00
FN	.00	.06	.00	.00	.00
count	27	457	83	43	
overlap	.68	.54	.67	.80	

b

ground truth / *detection*	bike	car/taxi	van	bus/lorry	FP
bike	.96	.03	.02	.00	.50
car/taxi	.00	.88	.12	.00	.05
van	.00	.02	.82	.00	.04
bus/lorry	.00	.01	.02	1.00	.03
FN	.04	.07	.02	.00	.00
count	28	361	57	29	
overlap	.73	.66	.70	.76	

c

Table 1 a) Confusion matrix for 3DHOG detector and classifier. b) FFT system performance c) Baseline algorithm (motion silhouette) from [3] evaluated on the same video data.

performance for histogram features is still reasonable (Recall 62.5% at precision 63.9%) considering the crude nature of the feature. This demonstrates the effectiveness of the patch extraction framework based on 3D interest points to deal with basic features. By using the more descriptive 3DHOG feature, improvement to this baseline approach is observed of recall 18.6% and precision 18.1%. For sensitivity analysis of the patch size, it is varied to $\delta = 0.8\,\mathrm{m}$ at resolution $\rho = 20\,\mathrm{P/m}$ and $\delta = 0.5\,\mathrm{m}$ at $\rho = 16\,\mathrm{P/m}$, which causes the classification performance to drop by 5% and 15% respectively[1].

7. Conclusions

A novel algorithm, 3DHOG, for detection and classification of road users in urban scenes was presented. This is an extension to HOG feature extraction by applying 3D spatial modelling to operate on still images and thus overcoming the reliability limitations of motion silhouettes. This single solution handles variable viewpoints for rigid vehicles as well as pedestrians. A training framework is proposed generating weights for learned interest points for classification. Three algorithms for features based on HOG, FFT and simple histograms are evaluated and show comparable performance to a baseline approach using motion silhouettes. The classifier sweeps the hypotheses space to find the best match

[1] Confusion matrixes for all evaluated cases are provided in the supplementary material

between images (observation) and 3D models based on the average match measure between interest points and the training data.

For future work, we have started working on the integration of the classifier with frame to frame tracking will provide the opportunity to demonstrate the full potential of the algorithm on partially occluded objects. Tracking predictions can be used as additional cues for classification (especially for example for turns) and conversely, classification can assist tracking.

Acknowledgements

We are grateful to the Directorate of Traffic Operations at Transport for London for funding this project on Classification of Vehicles and Pedestrians for Urban Traffic Scenes. The i-LIDS dataset provided by the UK Home Office is used for evaluation complying with the academic license.

References

[1] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European Conference on Computer Vision (ECCV)*, volume 3951 of *LNCS*, pages 404–17, 2006.

[2] Home Office Scientific Development Branch. Imagery library for intelligent detection systems i-lids. http://scienceandresearch.homeoffice.gov.uk/hosdb/cctv-imaging-technology/video-based-detection-systems/i-lids/ [accessed 19 December 2008].

[3] Norbert Buch, James Orwell, and Sergio A. Velastin. Detection and classification of vehicles for urban traffic scenes. In *International Conference on Visual Information Engineering VIE08*, pages 182–187. IET, July 2008.

[4] Norbert Buch, James Orwell, and Sergio A. Velastin. Urban road user detection and classification using 3d wire frame models. *IET Computer Vision [accepted]*, 2009.

[5] D. Crandall, P. Felzenszwalb, and D. Huttenlocher. Spatial priors for part-based recognition using statistical models. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 10–17, June 2005.

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893, 2005.

[7] Navneet Dalal, Bill Triggs, and Cordelia Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV 2006*, pages 428–441, 2006.

[8] S. Gupte, O. Masoud, R.F.K. Martin, and N.P. Papanikolopoulos. Detection and classification of vehicles. *Intelligent Transportation Systems, IEEE Transactions on*, 3(1):37–47, 2002.

[9] P. KadewTraKuPong and R. Bowden. An improved adaptive background mixture model for real-time tracking with shadow detection. In *Proceedings of 2nd European Workshop on Advanced Video-Based Surveillance Systems*, 2001.

[10] Z. Kim and J. Malik. Fast vehicle detection with probabilistic feature grouping and its application to vehicle tracking. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, volume 1, pages 524–531, 2003.

[11] Alexander Kläser, Marcin Marszalek, and Cordelia Schmid. A spatio-temporal descriptor based on 3d-gradients. In *British Computer Vision Conference BMVC 2008*, volume 2, pages 995 – 1004, 2008.

[12] B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool. Dynamic 3d scene analysis from a moving vehicle. In *Computer Vision and Pattern Recognition. CVPR '07. IEEE Conference on*, pages 1–8, June 2007.

[13] B. Leibe, K. Schindler, N. Cornelis, and L. Van Gool. Coupled object detection and tracking from static cameras and moving vehicles. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(10):1683–1698, Oct. 2008.

[14] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 878–885, 2005.

[15] J. Liebelt, C. Schmid, and K. Schertler. Viewpoint-independent object class detection using 3d feature maps. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, June 2008.

[16] David G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision 1999, ICCV 1999*, volume 02, page 1150–1157, Los Alamitos, CA, USA, 1999. IEEE Computer Society.

[17] Xiaoxu Ma and W.E.L. Grimson. Edge-based rich representation for vehicle classification. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1185–1192, 2005.

[18] Stefano Messelodi, Carla Maria Modena, and Michele Zanin. A computer vision system for the detection and classification of vehicles at urban road intersections. *Pattern Analysis & Applications*, 8(1-2):17–31, September 2005.

[19] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[20] Brendan Morris and Mohan Trivedi. Improved vehicle classification in long traffic video by cooperating tracker and classifier modules. In *AVSS '06: Proceedings of the IEEE International Conference on Video and Signal Based Surveillance*, page 9, Washington, DC, USA, 2006. IEEE Computer Society.

[21] A. Opelt, A. Pinz, and A. Zisserman. A boundary-fragment-model for object detection. In *Proceedings of the European Conference on Computer Vision*, pages 575–588. Springer-Verlag Berlin Heidelberg, 2006.

[22] Kiseo Park, Daeho Lee, and Youngtae Park. Video-based detection of street-parking violation. In *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition CVPR 2007*, 2007.

[23] Yan Pingkun, S.M. Khan, and M. Shah. 3d model based object class detection in an arbitrary view. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–6, Oct. 2007.

[24] Xuefeng Song and R. Nevatia. Detection and tracking of moving vehicles in crowded scenes. In *Motion and Video Computing. WMVC '07. IEEE W. on*, pages 4–4, 2007.

[25] J. Starck and A. Hilton. Spherical matching for temporal correspondence of non-rigid surfaces. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1387–1394, 17-21 October 2005.

[26] Zhaoxiang Zhang, Min Li, Kaiqi Huang, and Tieniu Tan. Boosting local feature descriptors for automatic objects classification in traffic scene surveillance. *International Conference on Pattern Recognition (ICPR) 2008*, 2008.

3D Extended Histogram of Oriented Gradients (3DHOG) for Classification of Road Users in Urban Scenes

Norbert Buch
norbert.buch@kingston.ac.uk
James Orwell
j.orwell@kingston.ac.uk
Sergio A. Velastin
sergio.velastin@kingston.ac.uk

Digital Imaging Research Centre
Kingston University
Kingston upon Thames, UK

8. Appendices

8.1. Logistics function for distance normalisation

A logistic sigmoid function is used to transform the distance measure d_k into a match measure m_k in the interval $[0,1]$. The function

$$m_k = \frac{1}{1+e^{a(b-d_k)}} \tag{1}$$

uses two parameters a and b for scale and shift of the transition region. The parameters can be estimated from the distance surface $\bar{\mathbf{D}}_{Mj}$ for every model. See Figure 1 for an example surface of an interest point. The values for this surface vary between interest points and therefore benefit from normalisation. The proposed parameterisation places the centre point of the distance surface d_C at the middle of the sigmoid function with a match measure of $m_k = 0.5$

$$\begin{aligned} m_k &= \frac{1}{1+e^{a(b-d_k)}}\bigg|_{d_k=d_C} = \frac{1}{1+e^{a(b-d_c)}} \\ \frac{1}{2} &= \frac{1}{1+e^{a(b-d_c)}}\bigg|_{b=d_c} = \frac{1}{1+e^{0}} \end{aligned} \tag{2}$$

which defines parameter $b = d_C$. The gradient of the sigmoid function at this point defines the other parameter a. The gradient should correspond to a line between the centre distance value at match measure $m_k = 0.5$ and the mean of all distance values $\bar{d}$ at match measure $m_k = 0$. This is illustrated as continuous blue line in Figure 2. The gradient of the line is

$$g = \frac{\bar{d} - d_C}{0.5} \tag{3}$$

which will be made equal to the gradient of the sigmoid function. The gradient of the logistics function at this point is

$$\left.\frac{dm_k}{dd_k}\right|_{\substack{d_k=d_C \\ b=d_C}} = \frac{-1}{\left(1+e^{a(b-d_k)}\right)^2} e^{a(b-d_k)}(-a)\Bigg|_{\substack{d_k=d_C \\ b=d_C}} = \frac{a}{2^2}, \tag{4}$$

which is made equal to the line gradient g:

$$\frac{a}{4} = g. \tag{5}$$

Using equation (3) and (5), the parameter a is given by

$$a = \frac{\bar{d} - d_C}{2}. \tag{6}$$

The resulting sigmoid function is illustrated in Figure 2. This function is used to convert distance measures $\mathbf{D}_{Mj}$ to match measures $\mathbf{M}_{Mj}$ during training and classification. An example output of the match measure $\mathbf{M}_{Mj}$ can be seen in Figure 3 showing a distinct peak, which is of the same height for all the interest points. By using the mean of all distance data points and therefore considering all data, a uniform drop of match measure is generated for different interest points. The impact of outliers is limited due to the bound output of the sigmoid function.

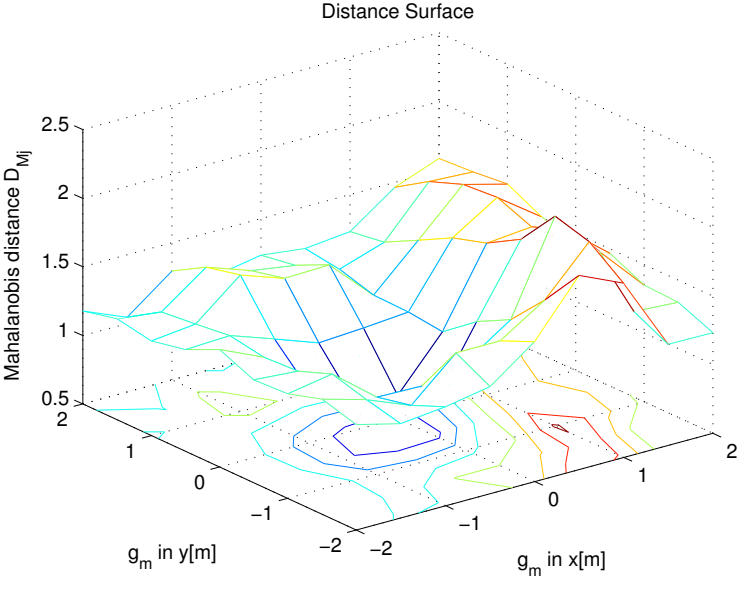


Figure 1 Example average feature distance surface. The centre at position $(0,0)$ corresponds to the training position and has usually the lowest value. The feature distance increases for coordinates further away from the centre.

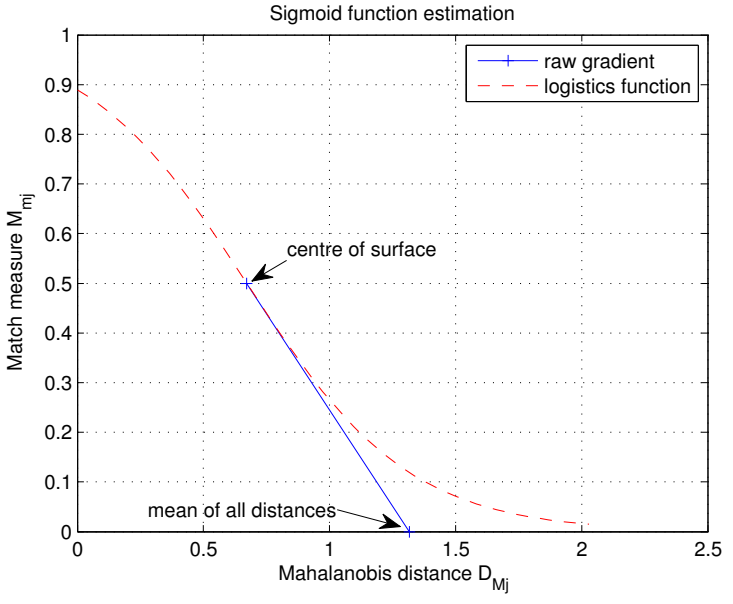


Figure 2 Estimated sigmoid function as dashed line. The continuous line is the gradient of the sigmoid function defined by the centre value of the distance surface and the mean distance of all grid points.

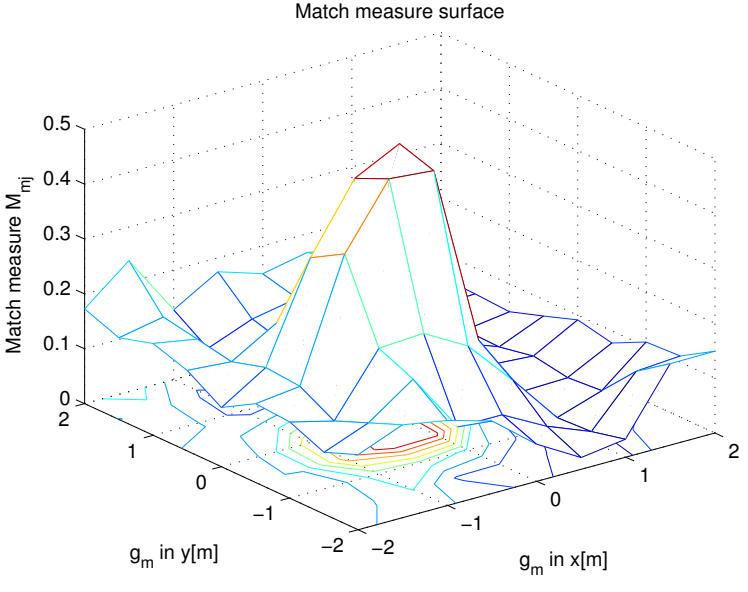


Figure 3 Final match measure surface after application of the sigmoid function. A distinct peak at the training position can be observed. This peak is set to the same value for all interest points.

8.2. Parameter influence

The influence of the 3D patch size is investigated. Reduced performance can be observed when reducing the patch to $\delta = 0.8\,\mathrm{m}$ at $\rho = 20\,\mathrm{P/m}$ and $\delta = 0.5\,\mathrm{m}$ at $\rho = 16\,\mathrm{P/m}$. Refer to Figure 4 for quantitative analysis.

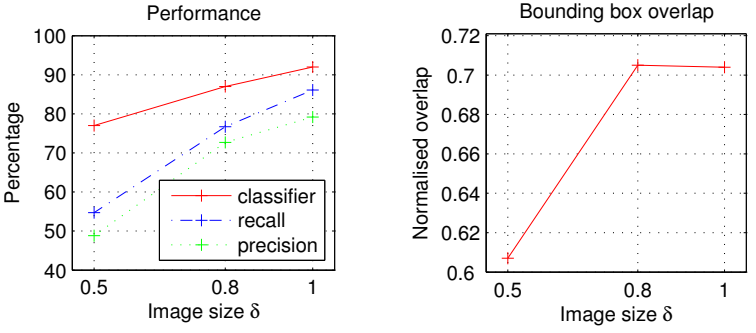


Figure 4 Performance increases in recall and precision for increased 3D patch size of 3DHOG classifier for oncoming driving direction.

8.3. Complete performance figures for three proposed features

This section provides all performance figures of the three algorithms proposed. An extended confusion matrix for full system performance, a confusion matrix of the classifier and a table with overall performance figures is given in Table 1 to Table 3.

ground truth / *detection*	bike	car/taxi	van	bus/lorry	FP
bike	1.00	.00	.00	.00	.44
car/taxi	.00	.83	.21	.03	.10
van	.00	.00	.67	.33	.08
bus/lorry	.00	.02	.02	.65	.00
FN	.00	.14	.10	.00	.00
count	27	361	48	40	
overlap	.70	.66	.73	.76	

ground truth / *detection*	bike	car/taxi	van	bus/lorry
bike	1.00	.00	.00	.00
car/taxi	.00	.97	.23	.03
van	.00	.00	.74	.33
bus/lorry	.00	.02	.02	.65
count	27	309	43	40

Symbol	*Value*
Recall R	81.1%
Precision P	82.0%
Classifier P_C	92.1%
Detector R_D	88.0%
Detector P_D	89.0%
GT Overlap	0.67

Table 1 Performance of 3DHOG feature. Left: full system performance; middle: classifier performance; right: cumulative figures.

ground truth / *detection*	bike	car/taxi	van	bus/lorry	FP
bike	1.00	.17	.34	.42	5.70
car/taxi	.00	.75	.39	.05	.32
van	.00	.02	.28	.07	.10
bus/lorry	.00	.00	.00	.47	.00
FN	.00	.06	.00	.00	.00
count	27	457	83	43	
overlap	.68	.54	.67	.80	

ground truth / *detection*	bike	car/taxi	van	bus/lorry
bike	1.00	.18	.34	.42
car/taxi	.00	.79	.39	.05
van	.00	.03	.28	.07
bus/lorry	.00	.00	.00	.47
count	27	430	83	43

Symbol	*Value*
Recall R	67.4%
Precision P	46.1%
Classifier P_C	70.5%
Detector R_D	95.6%
Detector P_D	65.4%
GT Overlap	0.57

Table 2 Performance of FFT feature. Left: full system performance; middle: classifier performance; right: cumulative figures.

ground truth / *detection*	bike	car/taxi	van	bus/lorry	FP
bike	.04	.04	.00	.03	.74
car/taxi	.04	.85	.61	.86	.31
van	.26	.04	.31	.00	.23
bus/lorry	.00	.00	.00	.11	.06
FN	.67	.08	.08	.00	.00
count	27	434	64	36	
overlap	.35	.58	.70	.67	

ground truth / *detection*	bike	car/taxi	van	bus/lorry
bike	.11	.04	.00	.03
car/taxi	.11	.92	.66	.86
van	.78	.04	.34	.00
bus/lorry	.00	.00	.00	.11
count	9	401	59	36

Symbol	*Value*
Recall R	69.9%
Precision P	57.9%
Classifier P_C	77.6%
Detector R_D	90.0%
Detector P_D	74.6%
GT Overlap	0.59

Table 3 Performance of histogram feature. Left: full system performance; middle: classifier performance; right: cumulative figures.