International Journal of u- and e- Service, Science and Technology
Vol.7, No.1 (2014), pp.159-176
http://dx.doi.org/10.14257/ijunesst.2014.7.1.15

Unsupervised Learning of Object Detectors for Everyday Scenes

Najeed Ahmed Khan[1] and David C. Hogg[2]

[1]*NED University of Engineering and Technology Karachi, Pakistan*
[2]*School of Computing University of Leeds, UK*
[1]*najeed@neduet.edu.pk*, [2]*d.c.hogg@leeds.ac.uk*

Abstract

This paper proposes an unsupervised learning framework in which models of objects' appearance classes are learned using their spatio and temporal information, from video. These models are used to detect objects of different classes in the everyday scene. The proposed technique combines appearance and motion features in a weighted combination framework resulting in models of object classes. Thus, better detection results are achieved compared to foreground based tracking and to those obtained in a supervised way. Since the proposed technique is unsupervised, a good detection rate is achieved without manual effort expended in data collection and labelling. Experimental results confirm that the proposed framework offers a promising solution for detection in unfamiliar scenes.

Key Words: *Unsupervised learning, Appearance classes, Object detectors*

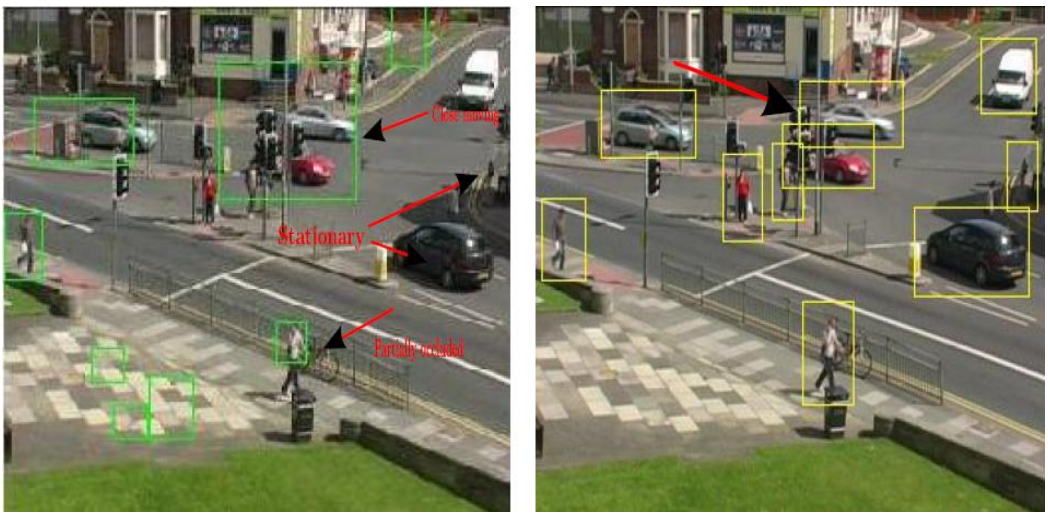
1. Introduction

The human ability to observe the world and learn object categories is remarkable. Objects such as cars, lorries, aeroplanes, and people are easily recognised and understood by humans as categories that have their unique logical divisions. This is despite the variances that are intrinsic to objects belonging to the respective categories. For example, objects of the same category can often be quite different with respect to features such as their appearance or behaviour. Even the same object may appear differently from different viewpoints and in different configurations (*e.g.*, postures). Furthermore, objects belonging to different categories can be confusingly similar with respect to their appearance or behaviour. One of the goals of computer vision is to reproduce this ability of humans to learn object categories and detect objects, despite the challenges arising from intra- and inter-class variances. Considerable research has been undertaken over the last few years to model the systems by which a machine can see and learn from observations. A number of fundamental problems have been addressed in this domain; for instance, object segmentation [3, 4, 5], object detection [6, 7, 8] scene analysis [9] and activity recognition [10, 11].

Training a machine to detect similar objects based on their appearance in a scene with minimal human intervention is an important and challenging research area. To deal with this challenge, various approaches for learning object classes have been proposed in the last few years. These approaches use different kinds of features of objects, such as appearance, motion, behaviour, affordance, and functionality for representing objects in a computer. These features have been used to learn object classes and/or to categorise objects in a fully supervised way [13, 15] or in a collaborative approach [14, 16] between human and computers. In recent work [2] adopted a discriminative approach using approximate hand annotations to learn a limb/non-limb classifier.

In comparison to the work on static images using an unsupervised learning approach [3, 5, 9, 18, 19] relatively less research has focused on learning from videos either for single object class [17] or multiple

ISSN: 2005-4246 IJUNESST



(a) (b)

Figure 1. Foreground Detection of Close Moving Objects and mis Detection of the Stationary Objects. (a) In Street Traffic Close by Moving Cars and the Stationary Objects in the Scene, Indicated by Arrows. Close Moving Objects are Detected as Single Foreground Object Regions. (b) Aim to Detect and Track these Foreground Object Regions SEPARATELY and also to DETECT STATIONARY OBJECTS in the SCEne, as INDICAted by ARROws

object classes [20]. Celik *et al.*, [20] proposed the idea of simultaneously training multiple detectors in an unsupervised way, using scene specific knowledge to guide the learning of object classes. They predefined the number of object classes and chose good training examples by using a predefined reference line in the image where the objects of interest have limited appearances. This approach may not be well suited for more challenging scenes. The framework proposed by Celik *et al.*, [20] is close to the approach proposed here. However, there are some significant differences which make our proposed approach more general. Firstly, rather than using just appearance based features, we combine it with motion and trajectory based features, since objects of the same class tend to have similar appearance and motion. Moreover, we learn the optimal combination of these features for any given scene in an unsupervised manner. This makes the system potentially more robust as it doesn't rely on appearance alone. The robustness of this combination has been demonstrated experimentally in Section IV. The second significant difference is that we not make any assumptions about the number or nature of the object categories in the scene, except that the objects are in motion.

Figure 1(a) shows example of detection of close moving objects as a single object (indicated by red arrows) by the tracker in [21]. This figure also shows the failure of detecting the stationary objects indicated by other red arrows. We aim to keep tracking such objects separately even when they are moving in close proximity. In addition, we also aim to detect objects even when they are stationary. Figure 1(b) is an example showing the separate detection of close moving objects and the detection of stationary object indicated by arrows, which is the goal of the research presented in this work.

The main motivation of the proposed technique in this paper is to combine the advantages of using foreground with that of trained object detectors, while minimising their respective disadvantages. We do not rely on previously trained object detectors. Instead, we use foreground extraction to obtain foreground blobs from a scene. We then use properties of the moving objects such as their appearance, motion, and trajectory features and learns appearance classes in an unsupervised way. In this manner, we significantly minimise the need for collection and labelling of training data for a new scene. We then train object detectors on these learned classes and proceed to detect

unseen objects for the same scene. Thus, we harness the power of trained object detectors with the information available from the foreground to create object detectors for a new scene in an unsupervised way.

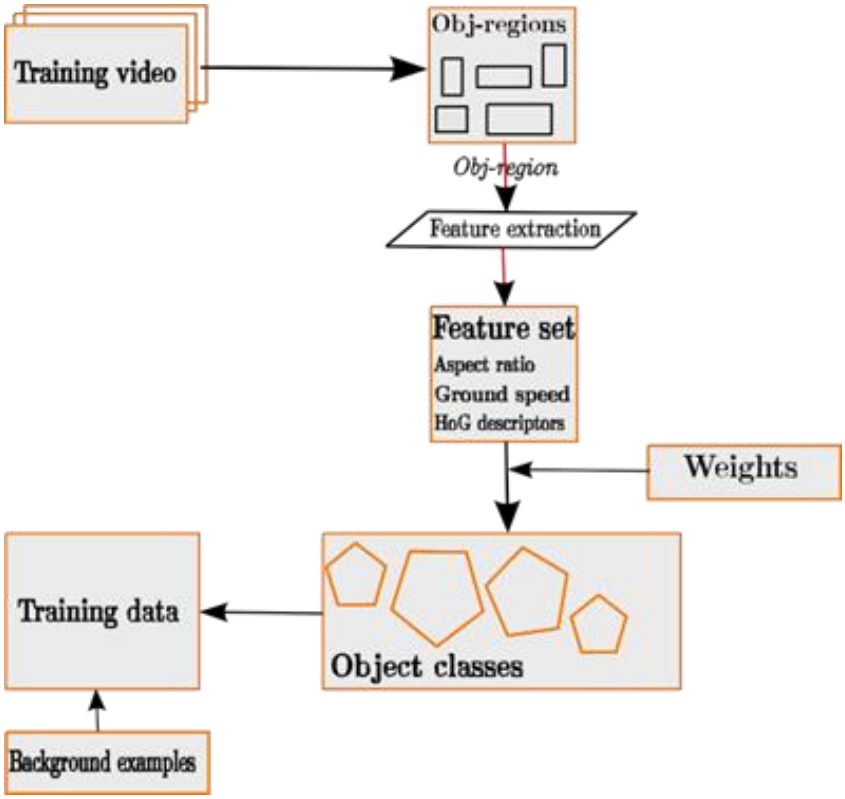


Figure 2. Flow of the System for Unsupervised Clustering of Foreground Object Regions to Acquire Training Examples from Unlabeled Video Sequences, where each Cluster Represents an Appearance Class. These Clusters are used to Train Detectors for each Appearance Class

The paper is organised into five sections, including this section. Section 2 describes the technique of unsupervised acquisition of training data to learn appearance classes. Section 3 describes the training of detectors used to detect objects in the observed scene. Section 4 describes the evaluation of the proposed framework. Finally, Section 5 concludes the paper by highlighting the propositions and future directions of our research work.

2. Unsupervised Acquisition of Training Data

Foreground segmentation using background subtraction and filtering noisy object regions from an input video is the first step of our system flow (Figure 2). From raw video frames acquired from a static camera, we segment foreground object regions using pixel-level background subtraction [21]. The collected foreground object regions are represented by a bounding box. Once we segment the foreground object region, we track it using the Nearest Neighbour Data Association algorithm [22], between consecutive frames.

Let $\mathcal{J} = (\tau_1, \tau_2, \ldots \tau_m, \ldots \tau_M)$ be the set of all object trajectories in an observed scene and 'τ_m' be a trajectory composed of the sequence of foreground object regions belonging to the same object, *i.e.*,

$$\tau_m = (o_1, \ldots . o_k \ldots . o_K) \quad (1)$$

where 'K_m' is the number of frames in which the object has appeared. Each foreground object region ok describes the object's attributes such as position, width and height, *i.e.*,

$$o_k = (x_k, y_k, w_k, h_k) \quad (2)$$

where (x_k, y_k) is the position (column, row address in image) and, w_k and h_k are the width and height respectively of a foreground object region in the K^{th} frame.

In outdoor scenes, the occurrence of noisy trajectories is not uncommon due to factors such as camera noise, variation in illumination and limitations of the background model. We automatically remove spurious that do not satisfy the following conditions:

$$|(x_k, y_k) - (x_{k-1}, y_{k-1})| < \lambda_{Disp} \qquad 1 < k \leq K_m \quad (3)$$

$$|(w_k, h_k) - (w_{k-1}, h_{k-1})| < \lambda_{Diff} \qquad 1 < k \leq K_m \quad (4)$$

$$K_m \geq \lambda_{Traj} \quad (5)$$

where λ_{Disp} and $\lambda_{Diff}$are predefined threshold and λ_{Traj} is the minimum accepted length of a trajectory.

Condition in Equation 3 ensures that there is no long displacement between frames. Condition in Equation 4 limits the deformation in width and height of object regions between frames. Condition in Equation 5 ensures only trajectories that persist over a given number of frames are kept.

2.1. Feature Extraction

The next step is the feature extraction, as shown in Figure 2, which is used to encode the objects. The object motion is computed in the form of ground speed and the appearance is represented using aspect ratio and a Histogram of Gradient (HoG) descriptor. Each of the individual features is then scaled by a given weight and concatenated into a final feature vector. Let s_k represent the displacement on the ground plane [30] of a foreground object between successive frames. Then the displacement is estimated as,

$$s_k = \left|\left(x'_k, y'_k\right) - \left(x'_{k-1}, y'_{k-1}\right)\right| \quad (6)$$

where $\left(x'_k, y'_k\right)$ are the coordinates on the ground plane corresponding to the centre base position of the foreground object region which is assumed in contact with the ground. An Homography mapping between the image plane and the ground plane is assumed knows. We compute the aspect ratio of each foreground object region using the width and height of the foreground object region. For a foreground object region $o = \langle x, y, w, h \rangle$ the aspect ratio is defined as,

$$r_k = \frac{w_k}{h_k} \quad (7)$$

The HoG descriptors for each foreground object region is computed in the observed scene. The HoG descriptors describe the object feature over the rectangular given patch or region of object. Therefore, HoG can be used to represent the rough shape [31] of the object of interest. To compute the HoG descriptors we adopt the Dalal and Triggs [23]

method. For a foreground object region the collection of HoG vectors is represented as h_k.

2.2. Combining Features

The combined feature vector is obtained by concatenating individual features for a foreground object region which is represented as,

$$\boldsymbol{f}_k^{\boldsymbol{\alpha}} = [\alpha_1.r_k \;\; \alpha_2.s_k \;\; \alpha_3.\boldsymbol{h}_k] \tag{8}$$

where the scalars $\leq \alpha = (\alpha_1, \alpha_2, \alpha_3)$ such that $\alpha_1 + \alpha_2 + \alpha_3 = 1$ and $0 \leq \alpha_1, \alpha_2, \alpha_3 \leq 1$. Finally, the matrix of combined feature vectors for all object regions are represented as,

$$\boldsymbol{F}^{\boldsymbol{\alpha}} = \begin{bmatrix} \boldsymbol{f}_1^{\boldsymbol{\alpha}} \\ \boldsymbol{f}_2^{\boldsymbol{\alpha}} \\ \cdot \\ \vdots \\ \boldsymbol{f}_T^{\boldsymbol{\alpha}} \end{bmatrix} \tag{9}$$

where the subscript now varies over all object regions and trajectories, and T is the total number of foreground object regions $= (T = (\sum_{m=1}^{M} K_m - 1)$, omitting first object region in each trajectory.

2.3. Clustering to obtain classes

Clustering is the next step after the acquisition of training data to obtain appearance classes, Figure 2, we cluster the matrix of combined feature vectors $\boldsymbol{F}^{\boldsymbol{\alpha}}$. In order to keep the classifier unsupervised we avoid using domain knowledge and do not fix the number of clusters. We cluster objects of training data with a varying numbers of clusters. The obtained set of clusters for input object set is represented as a partition of the T feature vectors:

$$\Theta = \{\theta_1, \theta_2, \theta_3, \ldots \ldots \theta_\eta\}, \tag{10}$$

where $\eta \leq T$ and θ can be given as,

$$\theta = \{o_j | j \in B\}, \tag{11}$$

where B is the set of objects regions belong to a cluster θ. The clustering feature vector $\boldsymbol{F}^{\boldsymbol{\alpha}}$ provides class labels for each foreground object region of the training data.

2.4. Parameter Estimation

The number of clusters η and the weights α are unknown and must be estimated from the training data in the matrix defined in Equation 9. We do this by maximizing an intrinsic measure of goodness for a given clustering Θ. To do this we use the Fisher's ratio [24], which measures the separability of clusters.

Let μ_θ and $\mu_{\theta'}$ be the means and σ_θ^2 and $\sigma_{\theta'}^2$ be the variances of combined features defined in Equation 9 of the objects belonging to clusters θ and θ' in Θ respectively. The Fisher ratio $z(\theta, \theta'))$ is defined as the ratio between the inter-class variance and the intra-class variance, which is given as:

$$z(\theta, \theta') = \frac{(\mu_\theta - \mu_{\theta'})^2}{\sigma_\theta^2 + \sigma_{\theta'}^2} \tag{12}$$

This definition is extended [24] to the case of multiple clusters, by taking the average Fisher ratio between all pairs:

$$Z(\Theta) = \frac{1}{|\Theta|(|\Theta|-1)} \sum_i \sum_{j \neq i} z(\theta_i, \theta_j) \quad (13)$$

The average Fisher ratio alone is not enough to compare the separability between the clusters where the number of clusters may vary. In order to compare clustering with different number of clusters we use a Minimum Description Length MDL-like principle [25] that scales the average Fisher ratio by a factor $|\Theta|^{\lambda}$ that penalises larger numbers of clusters, where $\lambda \geq 1$. We take the log of the product to obtain:

$$R(\Theta) = \log\big(Z(\Theta)\big) - \lambda \log(|\Theta|) \qquad \lambda \geq 1 \quad (14)$$

where $R(\Theta)$ is an MDL-like measure. Here, $\log\big(Z(\Theta)\big)$ corresponds to data-term and $\lambda \log(|\Theta|)$ corresponds to size of the model.

Using the parameters (α, η) defined above, Equation becomes

$$\mathrm{R}\big(\Theta_{\alpha,\eta}\big) = \log\Big(\mathrm{Z}\big(\Theta_{\alpha,\eta}\big)\Big) - \lambda \log\big(|\Theta|_{\alpha,\eta}\big) \qquad \lambda \geq 1 \quad (15)$$

where $\mathrm{R}\big(\Theta_{\alpha,\eta}\big)$ is an MDL-like measure for each value of the parameters (α, η). We determine the optimal values of the parameters (α^*, η^*) in two stages. First we determine top ranking optimal weights $\boldsymbol{\alpha}^*$. By top ranking optimal weights we mean the highest ranking value of $\boldsymbol{\alpha}$ across the numbers of clusters. To define $\boldsymbol{\alpha}^*$ we first define $\boldsymbol{\alpha}^*\eta$ for each value of η

$$\alpha^*_{\eta} = \underset{(\alpha)}{\mathrm{argmax}} \ \big\{\mathrm{R}\big(\Theta_{\alpha,\ \eta}\big)\big\} \quad (16)$$

Then for any α we define rank $o(\alpha)$ as the number of η for which α is maximum (that is those η for which $\alpha = \alpha^*_{\eta}$). Then we choose that α which has the highest rank, called top ranking α^*.

$$\boldsymbol{\alpha}^* = \underset{(\alpha)}{arg\max}\{O(\alpha)\} \quad (17)$$

Second having determined the optimal weights α^*, we find the optimal number of clusters η^*.

$$(\eta^*) = \underset{(\alpha)}{\mathrm{argmax}}\big\{\mathrm{R}\big(\Theta_{\alpha^*,\eta}\big)\big\} \quad (18)$$

In order to validate our unsupervised approach, we also compute optimal parameters with respect to category in ground truth, using the Rand Index [26]. These ground truth of the dataset are obtained in a semi-supervised way.

3. Training and Using Multiple Detectors

We train a bank of detectors consisting of one detector for each appearance class. The whole architecture of building a bank of detectors and using them to detect objects of interest may be divided into two phases: the training phase and the detection phase. The training phase creates a set of binary classifiers, one classifier for each appearance class. The detection phase uses the learned classifier(s) to detect objects of interest (if present) at multiple scale and positions in the frames of a test video.

3.1. Training Phase

The first step of training a bank of detectors c.f Figure 3, is the creation of training examples for each appearance class. We automatically select positive and negative training examples for each object class from the set of clusters Θ in addition to background examples. The selection of positive examples for each class is simple. Each cluster $\theta \epsilon \Theta$ is selected as a set of positive examples. The corresponding negative training examples include selected clusters from the training data plus background examples. We include examples from other clusters in the set of negatives in order to reduce confusion between classes.

We compute mean features of each object belonging to a cluster $\theta \epsilon \Theta$. if $\boldsymbol{F}_0^{\alpha^*}$ represents the optimal feature value of the foreground objects belonging to cluster θ, then the mean value of the features represented by $\bar{\mathrm{F}}_\theta^{\alpha^*}$ is estimated as,

$$\bar{\mathrm{F}}_\theta^{\alpha^*} = \frac{1}{|\theta|}\sum_{o\epsilon\theta} \mathrm{F}_o^{\alpha^*} \tag{19}$$

The L_2 norm Euclidean distance between the clusters $(\theta, \theta') \epsilon \Theta$ is defined as,

$$\mathrm{d}(\theta, \theta') = \left\|\bar{\mathrm{F}}_\theta^{\alpha^*} - \bar{\mathrm{F}}_{\theta'}^{\alpha^*}\right\| \tag{20}$$

For a set of positive examples belonging to cluster $\theta \epsilon \Theta$, the set of clusters consisting of negative training examples (other than the background examples), represented by $\widehat{N}_\theta$, are obtained as,

$$\widehat{N}_\theta = \underset{\theta' \epsilon\, \Theta}{U\theta'} \quad \text{such that } d(\theta, \theta') \geq \lambda_{Clst\_diff} \tag{21}$$

.

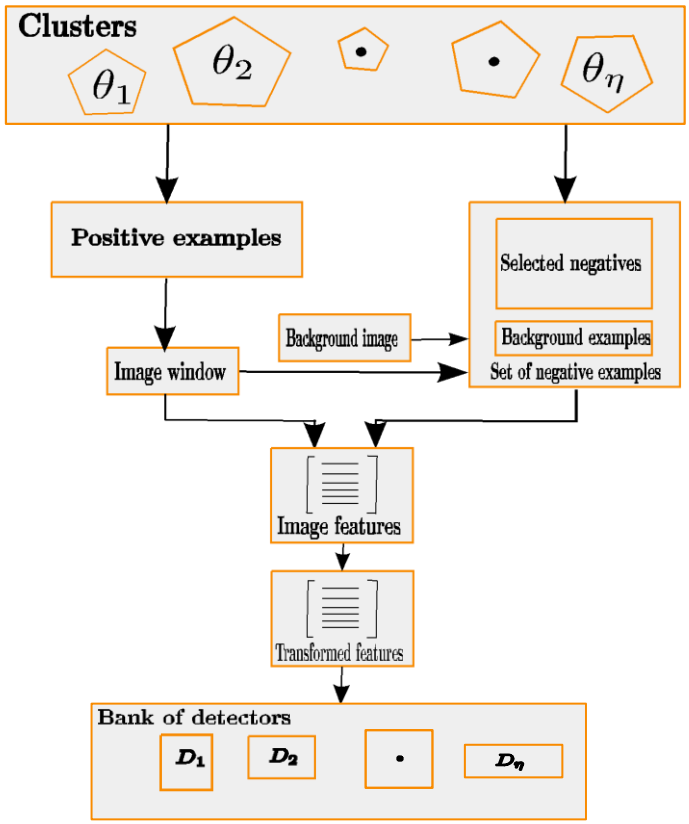


Figure 3. Training Phase: Train a BAnk of DETectors, Consisting as many Detectors as Number of CLusters, where each CLuster Corresponds to an Appearance Class

where $\lambda_{\text{Clst\_diff}}$ is the threshold. We estimate this threshold as the mean of the minimum and the maximum value in the matrix $d(\theta, \theta')$. We believe that this simple heuristic is also reasonable because the distance of set of positive examples belonging to a cluster θ is likely to be less than mean value. This implies that the set of examples belonging to a cluster θ' for which the distance is greater than the mean value would most likely be negative examples.

Finally, the set of training examples consisting of positive and negative training examples to train a detector is define below. Let Ω_θ represent a set training examples for an appearance class, then

$$\Omega_\theta = \{\theta, \hat{N}_\theta, \Gamma_\theta\} \tag{22}$$

where θ and $\hat{N}_\theta$ defined above, and Γ_θ is the set of background examples corresponding to the cluster θ. The set of training examples for other appearance classes in the observed scene are obtained in a similar way. Background examples are obtained from frames of the training video which do not contain any moving foreground objects. Some small supervision is involved in selecting background images that do not contain moving objects.

3.2. Object Representation

Training examples of each class are required to represent a fixed size image window. The same sized image window is used to detect objects of that class in test images or video. We automatically compute the fixed size image region window separately for each appearance class of the observed scene by using the mean width and height of the foreground objects regions belonging to the cluster. The fixed size image window for cluster θ corresponding to a class be represented by I_θ which is estimated as,

$$I_\theta = \bar{w}_\theta \times \bar{h}_\theta \tag{23}$$

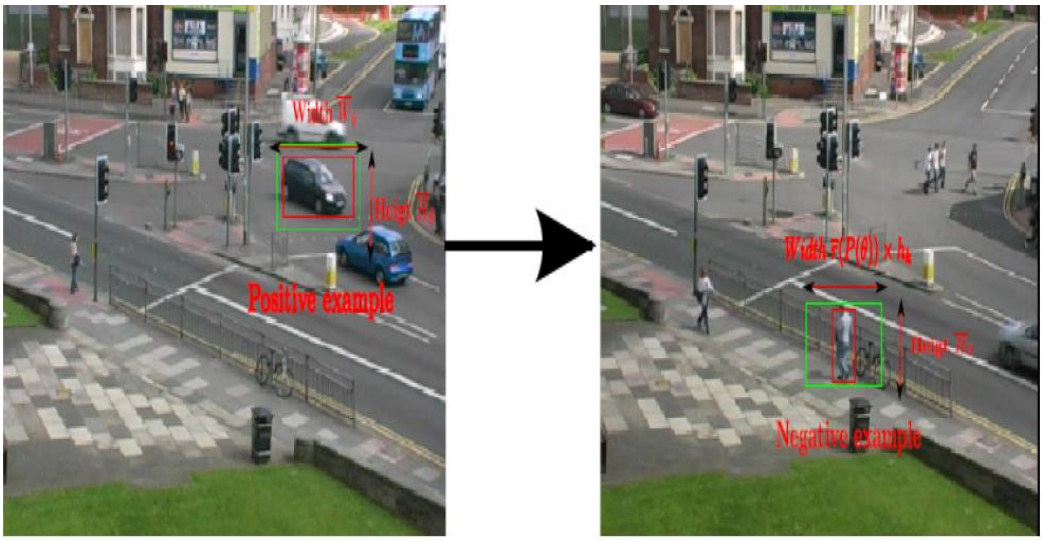


Figure 4. An Example of Computing a Same Aspect Ratio from a Negative Training Example Corresponding to a Positive Training Example, Sample Positive Training Examples (Visible Object Car) Belonging to a Class with Image Window Size [57, 36] Pixels and the Corresponding Negative Example with Actual Size Foreground Object Region Represented by Red Bounding Box and the Determined Mean Aspect Ratio of the Positive Examples (e.g, car) Represented by Green Bounding Box

Where

$$\bar{w}_\theta = \frac{1}{|\theta|} \sum_{\langle x_k, y_k, w_k, h_k, \rangle \, \epsilon \, \theta} w_k$$

$$\bar{h}_\theta = \frac{1}{|\theta|} \sum_{\langle x_k, y_k, w_k, h_k, \rangle \, \epsilon \, \theta} h_k$$

$$\bar{r}_\theta = \frac{\bar{w}_\theta}{\bar{h}_\theta}$$

Each of the positive training examples belonging to the cluster θ and the corresponding negative training examples are then re-scaled with respect to the fixed size image window I_θ. If the negative training examples are only the background examples then re-scaling them with respect to the fixed size image window I_θ is easy. However, when the negative training examples are the chosen clusters $\theta' \, \epsilon \, \hat{N}_\theta$ defined in Equation 21 containing a variety of object instances with varying object sizes, then extracting them with a fixed size image window, such that whole object instances enclosed in the image window I_θ, requires care. To extract such same sized negative examples, we maintain the aspect ratio of each negative example the same as the corresponding mean aspect ratio $\bar{r}_\theta$. Then we re-scale them with respect to the fixed size image window I_θ . To achieve this, we go back into the training images and pull out the negative training examples by enclosing objects of interest within a box with the aspect ratio equal to $\bar{r}_\theta$. The aspect ratio of a negative training example $o_k = \langle x_k, y_k, w_k, h_k, \rangle \, \epsilon \, \theta'$ where $\theta' \, \epsilon \, \hat{N}_\theta$ is estimated as,

$$r_k = \frac{w_k}{h_k} \tag{24}$$

To maintain the same aspect ratio we use the conditions:

$$if \quad r_k \geq \bar{r}_\theta \quad \text{then } w_k = \bar{w}_\theta \quad \text{and} \quad h_k = \frac{w_k}{\bar{r}_\theta}$$

$$if \quad r_k < \bar{r}_\theta \quad \text{then } h_k = \bar{h}_\theta \quad \text{and} \quad w_k = \bar{r}_\theta \times h_k$$

Figure 4 shows examples of visual results of the same aspect ratio of the negative example corresponding to the positive example.

The next stage in learning a classifier is feature extraction from each training example. The image feature extraction process maps image windows to a fixed size feature space which encodes visual foreground object regions for the classifier. We extract image features of each positive and negative (together with background) training example belonging to the training set Ω_θ . We use dense and overlapping histogram of oriented gradient HoG descriptors, proposed by Dalal and Triggs [23] to encode the visual objects regions.

The final stage of the learning phase is the training of a classifier that forms the basis of a detector, shown in Figure 3. We use a linear Support Vector Machine (SVM) as our baseline binary classifier, which proved to be the most reliable and scalable of the classifiers tested in our initial experiments. The extracted HoG features of each training example belong to the set of training Ω_θ for the appearance class corresponding to the cluster $\theta \, \epsilon \, \Theta$ are fed into the linear SVM. From the learning process we train a bank of SVM detectors, one detector for each appearance class. Each detector is trained for cluster $\theta \, \epsilon \, \Theta$ where each cluster θ corresponds to an appearance class.

3.3. Detection Phase

Our detection framework builds on the HoG detector presented by Dalal & Triggs [23]. Each detector in the bank is applied to search for the object of interest in the test video. The detectors scan the whole image (one detector at a time) at multi-scales with a (separate) fixed size image region window. Each detector computes image features with respect to the fixed size image window, and makes decisions of object/non-object for which it searches, based on some predefined threshold.

We evaluate the performance of the bank of detectors compared to the ground-truths and the foreground tracker.

4. Experiments and Evaluation

We evaluate the performance of our proposed method on a street traffic dataset, containing multiple object categories. The Street-traffic [32] dataset includes several classes of moving objects (people, cars, buses, and trucks). The statistics of dataset is given in table I(A).

Table I. Statistics Street-traffic dataset: (a) Output from the foreground tracker (b) Ground truth

Sequence: Street-traffic	
Number of frames	28891
Duration	19 m 15s
Total number of objects	11306 instances
Cars	5632 instances.
People	1672 instances.
Buses	926 instances.
Mix of other objects (people, cars, busses)	3076 instances

Sequence: Street-traffic	
Number of frames : 2799	
Object class	Number of instances
Cars	902
People	1109
Buses	224
Total objects	2235

To evaluate the performance of the trained detector we also collect ground truth from test videos, which are used to verify the prediction of content in the test images. Table I(B) summaries the ground truth for each object category in the training dataset.

Table II. Summary Clustering based on F-Ratio Measure. In the Table Column one Represents the Varying Number of Clusters, each Cell in Columns 2 Represent the Optimal F-Ratio Measure and the Corresponding Optimal Weights

Summary for clustering evaluation using F-ratio Street-traffic dataset	
No of Clusters	Optimal F-Ratio Optimal weights $[\alpha_1, \alpha_2, \alpha_3]$
2	0.328 [0.4 0.1 0.5]
3	0.413 [0.4 0.1 0.5]
4	0.375 [0.5 0.2 0.3]
6	0.345

	[0.4 0.2 0.4]
10	0.269 [0.4 0.2 0.4]
14	0.22 [0.4 0.1 0.5]

A. Baseline: Clustering Foreground Regions using HoG Features Only

In this experiment we report the results obtained by clustering foreground object regions using HoG features only. This approach has been adopted by Celik et al. [20] and hence forms a suitable baseline for the approach proposed here.

To evaluate the baseline approach, i.e. Celik's approach, we fixed parameters of our framework, $\eta = 3$ (same as the number of objects category in dataset), weights $\alpha = [0, 0, 1]$ (corresponds to the HoG feature in our framework) to form appearance classes by clustering foreground regions using only HoG features.

The learned clusters are evaluated in two ways, (i) with respect to ground truth of clusters using Rand index, (ii) with respect to performance on detection. These results provide a baseline with which we compare our approach below.

The clustering results are tabulated in Table III. This table shows that the Rand index is 49.6% based on HoG descriptors. However, the Rand index is 64.5% based on our proposed approach. This indicates the clustering performance using our proposed approach is significantly higher than Celik's approach. The detection performance of the SVM detector trained on clusters (obtained using Celik's approach) is shown in Figure 8(b). The maximum precision and recall is shown 0.45 and 0.23 respectively which is comparatively much less than the detection performance of a bank of detectors trained on the clusters obtained using our proposed approach shown in Figure 8(a).

B. Finding Optimal Clusters and Corresponding Parameters

One of the central aspects of this paper is a framework in which foreground objects could be clustered with different sets of parameters that correspond to (i) different weighted feature combinations with weights $\alpha = (\alpha_1, \alpha_2, \alpha_3)$; (ii) different number of clusters h. A technique for finding the optimal set of parameters in an unsupervised way using an MDL-like approach (as given in equation 15) was described in section II-A. The clusters corresponding to this set of parameters are used to train a bank of detectors. The following describes the experimental results.

In order to find optimal weights α^*, we find $\alpha^*\eta$ varying numbers of clusters η = 2,3,4,6,8,10 and 14.Using Equation 16 the optimal value of α for each number of cluster is tabulated in Table II. Using these values find rank $O(\alpha)$ for all α whenever $O(\alpha)$ is non zero. The values $O(\alpha)$ are plotted against α,

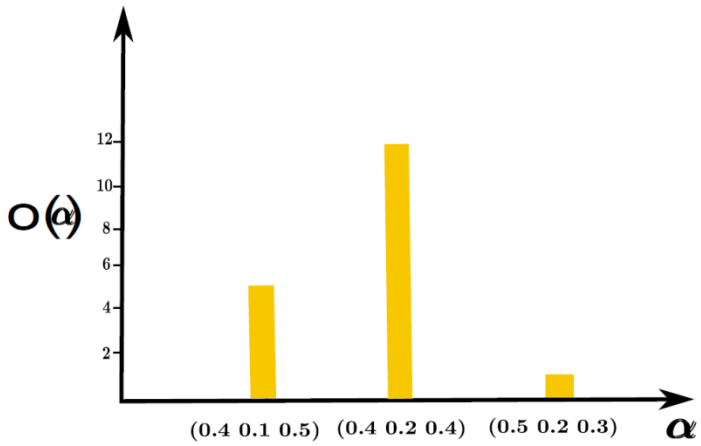


Figure 5. Rank of Weights $\alpha = (\alpha_1, \alpha_2, \alpha_3)$. The Highest Rank Corresponds to α^* = (0.4, 0.2, 0.4)

cf. Figure 5. Then using Equation 17 the optimal weight α^* is obtained as (0.4, 0.2, 0.4). Having determined α^*, we now compute η^* as those values of η that gives maximum F-ratio amongst all considered values of η tabulated in Table II. Thus, corresponding to the largest F-ratio measure **0.413** shown in the table, the optimal values of the parameters are: weights $\alpha^* = (\alpha_1, \alpha_2, \alpha_3)$ = (0.4, 0.2, 0.4) and the number of clusters η^* = 3.

It is intuitive that for a particular domain, certain features are more important than other features for the task of discriminating between objects of different appearance classes. The weights capture the relative importance of these different features. Globally optimal weights are specific to the domain for which they are obtained. In a new domain, these globally optimal weights may be different. We would like to find a fixed set of weights that are likely to be effective across different numbers of clusters as appropriate for a new domain. This is the motivation for choosing the most repeated weights (top ranked). Another motivation arises due to the variability of clustering produced by k-means. Choosing a globally consistent (most repeating) weights across different number of clusters would help address this problem also.

C. Validation of Parameters using Ground Truth

In the following, we validate the parameters learned above in an unsupervised way, with those learned by using the ground truth. We find optimal parameters w.r.t the category in ground truth on a single dataset using Rand index. From the experimental results, shown in Table III, the optimal values of parameters are computed using Rand index. Thus corresponding to the optimal clustering the Rand index measure 64.5% at the optimal values of the parameters: weights $\alpha^* = (\alpha_1, \alpha_2, \alpha_3)$ = (0.4, 0.2, 0.4) and the number of clusters η^* = 3. These set of parameters values are the same as those obtained using our unsupervised framework. Thus they validate the optimal values obtained in an unsupervised way. We notice that using the K-Means clustering algorithm the optimal weights vary slightly. For this reason, $(\alpha_1, \alpha_2, \alpha_3)$ = (0.5, 0.1, 0.4), (0.4, 0.1, 0.5), (0.3, 0.2, 0.5) can also be sets of good weights for combining the aspect ratio, ground speed and HoG descriptor features respectively.

Training and Detection: Having learned an optimal clustering of foreground regions, we regard them as learned appearance classes and use them for training appearance object class detectors. We train a bank of detectors consisting one detector for each appearance class, using a linear SVM classifier as they are regarded as fast and inherently robust to outliers [28]. We extract HoG features [23] with fixed size feature vectors for the SVM pattern recognition classifier. To detect object of interest, we apply each detector from the

bank of detectors to search for the object of interest in each image of a test video. Each image is scanned with a fixed size detection window at multiple scales for each detector in a bank.

Table III. Optimal Clustering based on each Single Feature and the Combined Features Set, Optimal Measures is the Rand Index

Summary for clustering $\eta = 3$ clusters Street-traffic dataset		
Weights	Feature set	Optimal Rand index
[1, 0, 0]	Aspect ratio	56.7
[0, 1, 0]	Ground speed	52.9
[0, 0, 1]	HoG descriptors	49.65
[0.4, 0.2, 0.4]	Weighted combination	64.5

4.1. Evaluation of Bank of Detectors

The performance of each bank of detectors is evaluated by a separate precision-recall curve. A predicted bounding box is considered correct (true positive) if it overlaps more than 50% with a ground truth bounding box, based on Pascal VOC'09 criteria [27] otherwise the detection bounding box is considered as false positive detection. We apply each detector from the bank of detectors to search for the object of interest in each image of test video. Each image is scanned with fixed detection window at multiple scales, as many times as we have the number of detectors in a bank.

Qualitative evaluation: We first summarise the detection results in a qualitative manner. Figures 6 and 7 show the detection performance of a bank of detectors consisting of one detector for each appearance class, compared to the [21] foreground detector performance and the ground truth. The detections found by the SVM detectors are shown with yellow color bounding boxes, whereas the ground truth and foreground detections are shown by red and green bounding boxes respectively. A qualitative inspection shows better performance with our learned detection compared to the foreground detector, which is our baseline. One advantage that arises naturally with the use of the learned detectors in comparison to the foreground detector is that the stationary object instances are comprehensively detected by the learned detector, whereas the foreground detector tends to miss objects when they are stationary beyond certain duration. Another observation is that the learned detectors are more capable of detecting closely moving objects.

Quantitative evaluation: The detection performance of the bank of detectors using our proposed approach and our baseline approach, trained for the street traffic dataset is also shown by precision recall curve in Figures 8(a) and 8(b) respectively. Figures 8(a) indicates the maximum precision recall are 0.92 and 0.8 respectively for the bank of detectors obtained using our proposed approach. However the maximum precision recall are only 0.45 and 0.23 respectively for the detector obtained using our baseline approach shown in Figures 8(a)

We also compare the detection performance of the bank of detectors obtained using our proposed approach with the detections of foreground detector [21] used to collect foreground objects. Figure 8(a) indicates the maximum precision recall are 0.29 and 0.072 respectively which is significantly low than the precision recall values of learned detectors obtained using our proposed approach.

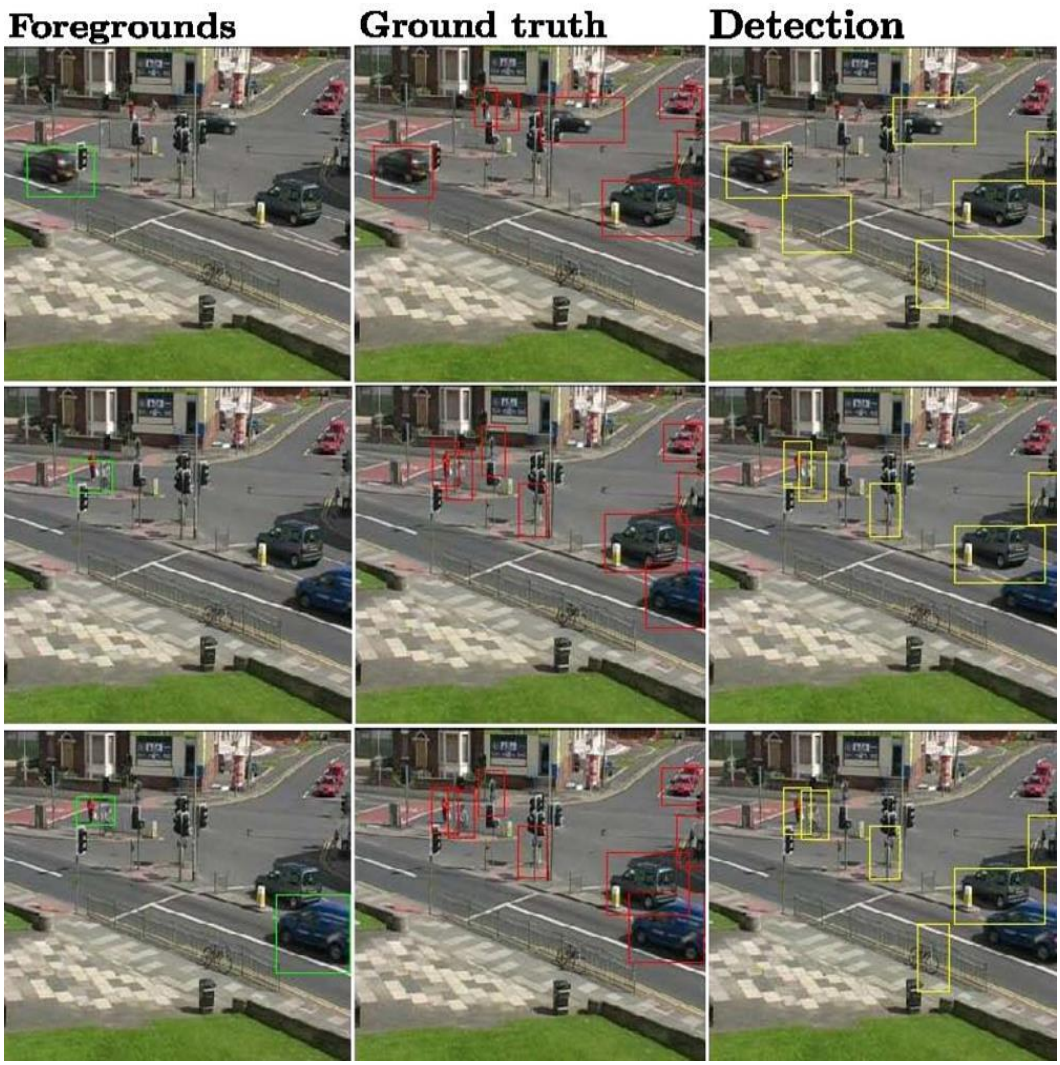


Figure 6. Some Examples of Detections on the Street-traffic Test Images for Final Detections of Instances. The First Column shows the Foreground Detection, Second Column shows Ground Truth Bounding Boxes and the Third Column Shows the Detections Found by the SVM Detectors of the Bank. Column Three shows that the Trained SVM Detectors have a Capability of Separately Detecting Close Moving Instances

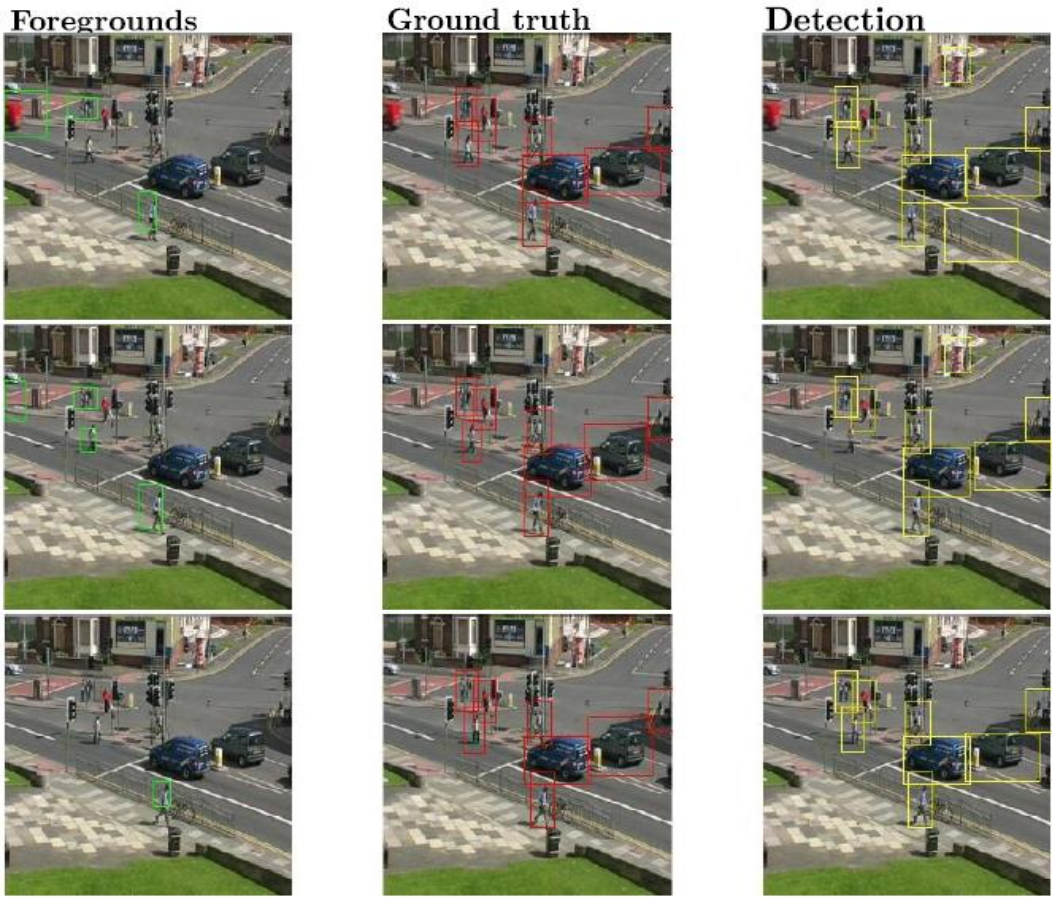


Figure 7. Some Examples of Detections on the Street-traffic Test Images for Final Detections of Instances. The First Column shows the Foreground Detection, Second Column shows Ground Truth Bounding Boxes and the Third Column shows the Detections Found by the SVM Detectors of the Bank. Column Three shows that the Trained SVM Detectors have a Capability of Separately Detecting Close Moving Instances

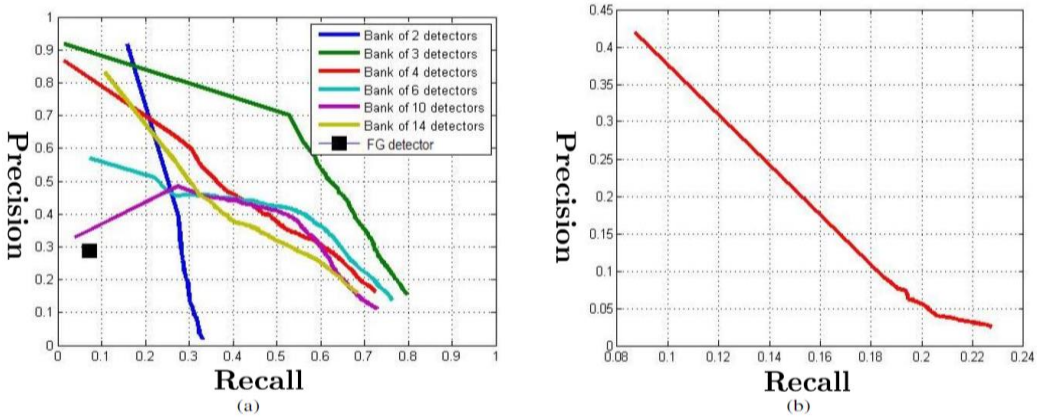


Figure 8. (a) Recall-precision Curves showing the Detection Performance of each Bank of SVM Detectors. SVM Detectors are trained on Positive Examples versus Selected Negative Examples in Addition to the Background Examples. (b) Recall-precision Curve Showing the Detection Performance, the SVM Detector is trained on HoG based Clusters. This is our Baseline Approach

5. Conclusion

The proposed framework in this paper incorporates object features such as aspect ratio, ground speed, and HoG descriptors. The experimental results show that a combined feature set provides better clustering results, and in particular that performance is superior to the use of HoG features alone. Thus we conclude that the learned detector significantly outperforms the baseline approach (i.e. Celik's approach) and the foreground detector. An important conclusion is that the combined features - aspect ratio, ground speed and the HoG - provide better clusters of the objects compared to the clusters obtained by using the features individually. Moreover, the optimal weights $(\alpha_1, \alpha_2, \alpha_3)$ = (0.4, 0.2, 0.4) show that the features Aspect ratio and HoG descriptors, have similar significance, and are more significant than the feature Ground speed. This is intuitively reasonable, since this scene consists of several signals which tends to slow down the vehicles, thus lowering the ground speed. This explains why HoG features and aspect ratio may be better suited to form more distinct clusters. Figure 8(a) also shows that the bank of detectors trained for the Street-traffic dataset has optimal performance for three detectors. Each detector in the bank corresponds to an appearance class in the observed scene. The Street-traffic dataset has three optimal appearance classes which validated the optimal number of clusters obtained using F-ratio.

The proposed framework learns the appearance models for different object classes in the observed scene, as we have shown for the Street-traffic dataset. This learning is automatic and would take place for every new scene and indeed could be used to adapt to long-term changes within a scene. However, our framework may not work for object classification in a far field video [29], especially for small-sized object detection, for which the proposed appearance features may not be computed due to inadequate resolution.

Future work: The research work presented in this paper can be further expanded into different interesting scenarios. For example, learning the parameters automatically for noise filtering from the unlabelled data would facilitate the framework's applicability. Currently, in our approach, some supervision is involved. Further research can be pursued to improve the clustering method, which can automatically split and merge clusters, in order to obtain the optimal number of clusters for each dataset.

References

[1] J. Charles and M. Everingham, "Learning shape models for monocular human pose estimation from the Microsoft Xbox Kinect'', Proceedings of the 13th International Conference on Computer Vision-Workshop, Barcelona, Spain, **(2011)** 6-13 November.
[2] S. Johnson and M. Everingham, "Learning effective human pose estimation from inaccurate annotation", Proceedings of the 24th IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, USA, **(2011)** June 20-25.
[3] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation", Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no 8, **(2000)**, pp 888-905.
[4] M. Rubinstein, A. Joulin, J. Kopf, and Ce Liu, "Unsupervised Joint Object Discovery and Segmentation in Internet Images", Proceeding of the IEEE Conf. on Computer Vision and Pattern Recognition, **(2013)** June.
[5] B. C. Russell, W. T. Freeman, A. A. Efros, J. Sivic and A. Zisserman, "Using Multiple Segmentations to Discover Objects and their Extent in Image Collections", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, **(2006)**, pp. 1605-1614.
[6] C. P. Papageorgiou, M. Oren and T. Poggio, "A general framework for object detection", (article) Sixth International Conference on Computer Vision, **(1998)**.
[7] M. Weber, M. Welling and P. Perona, "Towards automatic discovery of object categories", Proceeding of IEEE conference on Computer Vision and Pattern Recognition, Hilton Head, SC, USA, **(2000)** June 13-15.
[8] M. Weber, M. Welling and P. Perona, "Unsupervised Learning of Models for Recognition", In Proceeding of the European Conference on Computer Vision-Part I, Ireland, **(2000)**.
[9] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, **(2005)**, pp. 524-531.

[10] X. Wang, K. Tieu and E. Grimson, "Learning Semantic Scene Models by Trajectory Analysis", Proceeding of the European Conference on Computer Vision-Part I, Graz, Austria, **(2006)** May 7-13.
[11] M. Sridhar, A. G. Cohn and D. C. Hogg, "Learning Functional Object-Categories from a Relational Spatio- Temporal Representation", Proceeding of 18th European Conference on Artificial Intelligence, Patras, Greece, **(2008)** July 21-25.
[12] T. F. Cootes, C. J. Taylor, D. H. Cooper and J. Graham, "Training models of shape from sets of examples", Proceeding of British Machine Vision Conference, Leeds, UK, **(1992)** September.
[13] P. Viola and M. Jones, "Rapid Object Detection Using a Boosted Cascade of Simple Features", Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, Kauai, HI, USA, vol. 1, **(2001)** December 8-14.
[14] O. Javed, "Online detection and classification of moving objects using progressively improving detectors", Proceeding of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, **(2005)** June 20-26.
[15] T. F. Cootes, C. J. Taylor, D. H. Cooper and J. Graham, "Training models of shape from sets of examples", Proceeding of British Machine Vision Conference, Leeds, UK, **(1992)** September, pp. 9-18.
[16] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training", Proceeding of the Eleventh Annual Conference on Computational learning theory, Madison, Wisconsin, USA, **(1998)** July 24-26, pp. 92-100.
[17] A. M. Baumberg and D. C. Hogg, "Learning flexible models from image sequences", Proceeding of 3rd European Conference on Computer Vision, Stockholm, Sweden, **(1994)** May 2-6, pp. 299-308.
[18] T. Hofmann, "Probabilistic latent semantic indexing", Proceeding of the 22nd annual international ACM SIGIR Conference on Research and development in information retrieval, ACM New York, NY, USA, **(1999)** August 15-19, pp. 50-57.
[19] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman and W. T. Freeman, "Discovering objects and their location in images", Proceeding of the IEEE 10th International Conference on Computer Vision, Beijing, China, vol. 1, **(2005)** October17-20, pp. 370-377.
[20] H. Celik, A. Hanjalic and E. A. Hendriks, "Unsupervised and simultaneous training of multiple object detectors from unlabeled surveillance video", Comput.Vis. Image Underst, vol. 3, **(2009)**, pp. 1076-1094.
[21] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction", Pattern Recognition Letters, vol. 27, **(2006)**, pp 773-780.
[22] S. S. Blackman and R. Popoli, "Design and analysis of modern tracking systems", Artech House, **(1999)**.
[23] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection", Proceeding of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, vol. 1, **(2005)** 20-26 June, pp. 886-893.
[24] K. Z. Mao, "RBF neural network center selection based on Fisher ratio class separability measure", IEEE Transactions on Neural Networks, vol. 3, **(2002)**, pp. 1211-1217.
[25] J. Rissanen, "An introduction to the MDL principle", Helsinki Institute for Information Technology, Tampere and Helsinki Universities of Technology, Finland, and University of London, England, **(2006)**.
[26] W. M. Rand, "Objective Criteria for the Evaluation of Clustering Methods", Journal of the American Statistical Association, vol. 66, **(1971)**, pp. 846-850.
[27] M. Everingham, L. Van Gool, C. Williams and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2009", Results (article), Citeseer, **(2009)**.
[28] V. N. Vapnik, "The nature of statistical learning theory", Springer Verlag, **(2000)**.
[29] B. Bose and E. Grimson, "Improving Object Classification in Far-Field Video", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, **(2004)**, pp. 181-188.
[30] A. Panagiota, K. Dimitrios and P. J. Stavros, "Detecting abnormal human behaviour using multiple cameras", Signal Process, vol. 89, **(2009)**.
[31] K. Onishi, T. Takiguchi and Y. Ariki, "3D human posture estimation using the HoG features from monocular image", Proceeding of the International Conference on Pattern Recognition, Tampa, Florida, USA, **(2008)** December 8-11, pp. 1-4.
[32] K. Najeed and D. C. Hogg, "Unsupervised learning of object detectors for everyday scene", PhD thesis, University of Leeds, UK, **(2011)**.

Authors

Dr. Najeed A. Khan was born in Pakistan 1966. **Najeed** received an MSc degree of computer science in 1996 from NED University of Engineering & Technology Karachi, Pakistan and a PhD degree from University of Leeds, UK in 2011.The major area of interest is computer vision. **Najeed** is now an Associate Professor of Artificial Intelligence at NED University Karachi, Pakistan and Ag Executive Officer at Centre of Software Research and Development (CSRD) endowment of Ministry of Science & Technology, Pakistan.

Dr. Najeed is a senior member of IACSIT.

David Hogg has been a Full Professor at the University of Leeds since 1990, where he now heads the Computer Vision group. He was head of the School of Computing from 1996 to 1999, and a Pro-Vice-Chancellor of the University from 2000 to 2004. During 1999-2000 he was a visiting professor at the MIT Media Lab in Cambridge. He is a member of the EPSRC College, a Fellow of ECCAI, an Associate Editor of IEEE-PAMI, has been on the programme committee for most of the leading international conferences in the field for over ten years and advises many research funding agencies worldwide on a regular basis.

