Extending the Performance of Human Classifiers using a Viewpoint Specific Approach

Endri Dibra
ETH Zurich
endri.dibra@inf.ethz.ch

Jerome Maye
ETH Zurich
jerome.maye@mavt.ethz.ch

Olga Diamanti
ETH Zurich
olga.diamanti@inf.ethz.ch

Roland Siegwart
ETH Zurich
roland.siegwart@mavt.ethz.ch

Paul Beardsley
Disney Research Zurich
pab@disneyresearch.com

Abstract

This paper describes human classifiers that are 'viewpoint specific', meaning specific to subjects being observed by a particular camera in a particular scene. The advantages of the approach are (a) improved human detection in the presence of perspective foreshortening from an elevated camera, (b) ability to handle partial occlusion of subjects e.g. partial occlusion by furniture in an indoor scene, and (c) ability to detect subjects when partially truncated at the top, bottom or sides of the image. Elevated camera views will typically generate truncated views for subjects at the image edges but our viewpoint specific method handles such cases and thereby extends overall detection coverage.

The approach is - (a) define a tiling on the ground plane of the 3D scene, (b) generate training images per tile using virtual humans, (c) train a classifier per tile (d) run the classifiers on the real scene. The approach would be prohibitive if each new deployment required real training images, but it is feasible because training is done with a virtual humans inserted into a scene model. The classifier is a linear SVM and HOGs. Experimental results provide a comparative analysis with existing algorithms to demonstrate the advantages described above.

1. Introduction

Detecting people in scenes via a fixed camera has been of great interest to the Computer Vision community in the recent years, due to the multitude of their applications, and has been extensively studied in the literature (e.g. [6, 1, 19, 24]). Many of these works succeed at detecting people in an open scene viewed from a generic viewpoint far away from the people. However, their performance typically suffers in more complicated cases; a human may not be detected when (a) occluders are present in the scene, (b) he/she is only partially visible (e.g. due to lying partly across an edge of the image) or (c) his/her appearance is significantly skewed due to severe perspective foreshortening caused by an oblique placement of the camera.

In this work, we propose viewpoint specific human detection to attack these problems. By 'viewpoint specific' detection we refer to a detection scheme where different classifiers are trained for different parts of the image, thus accounting for different human appearances across the image as well as truncations and occlusions by fixed occluders; such an approach exploits more contextual information about the scene, which can greatly improve the performance of a classifier under the adverse conditions mentioned above.

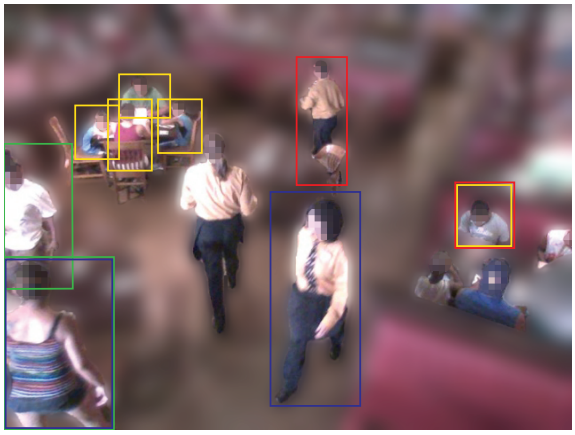


Figure 1. Difficult cases for human detection, handled by our method. Blue: Varying appearance due to perspective foreshortening. Green: Truncation close to the image boundaries. Red: Occlusion by static occluders. Yellow: Multiple people sitting close to each other. See text for more details. Mosaicing and blur have been applied for anonymization and confidentiality purposes.

Our test case is a restaurant scenario where both upright (standing/walking) as well as seated people are detected amidst a fixed furniture setting. This scenario exhibits all of the issues mentioned above: (a) the furniture frequently occludes the people, (b) humans are often truncated at the image boundary, but still need to be detected in order to extend the system coverage[1], and finally (c) the camera captures a wide angle and is placed at an elevated position, thus the image appearance of people in different locations in the scene varies greatly. We show examples of these problematic cases in Fig. 1.

Our method assumes a known 3D model of the scene, consisting of a coarse room map and approximate models of the contained scene objects. We divide the ground plane of the scene into a grid of overlapping square tiles, and train a dedicated human classifier at each tile using synthetic images. Each classifier is specific to the projection of its corresponding tile in the image plane. At run-time, all the classifiers are run on the real camera imagery to provide human detection at each tile. Naturally, capturing real image data for the training of each of these classifiers would be prohibitively time-consuming. We therefore avoid this requirement by inserting virtual humans (3D human models) and placing them with physically correct behavior (e.g. walking on the ground plane or seated at a table) in the scene, thus obtaining photo-realistic images of the scene. This task is automated, thus enabling us to reading generate as many training images as needed.

In this paper, we demonstrate human detection that accounts for scene occluders, perspective foreshortening, and partial views of humans. The presence of specialized classifiers for each 3D location provides explicit control of the training data at different scene locations and thus accounts for variations in appearance; at the same time, the modeling of the occluders (furniture) allows for detection of specific activities at a given location (such as sitting at a table) which would be impossible to do with a generic classifier.

Contributions: The contributions of the paper are - (a) the description of a complete system for human detection, (b) detection of two behaviors, standing and sitting, in a complex environment with scene occluders, (c) a comparative analysis of the system with state-of-the-art algorithms for human detection.

Structure of paper: Section 2 describes related work; Section 3 describes the setup of the 3D scene and how the 3D information is utilized to support human detection; Section 4 describes our method; and Section 5 contains results and a thorough comparison with state-of-the-art methods.

[1]Requirement (b) can be solved by having more cameras, but economic constraints limit this, while aesthetic constraints are especially important in a restaurant.

2. Related Work

In a detection task, the general paradigm to find a target in an image is based on a sliding window multi-scale search which is simple and effective, but also open to improvement. Recent works such as Alexe et al. [1] search for objects more intelligently based on context, utilizing few window evaluations. Our approach searches only at selected image locations and scales, reducing drastically the number of windows evaluated. Other works like [8, 19, 22, 23] explore the idea of using classifier grids in combination with online learning. These grids are placed on the 2D image and adaptive classifiers are trained online through boosting. Different from them, our grids are placed on the 3D scene. The works in [15] and [25] also make use of scene-specific knowledge although without using a classifier grid. Sudowe et al. [24] describe an efficient sliding-window object detection algorithm that incorporates ground plane constraints directly into the detector computation.

Detection methods can be broadly divided into *silhouette* and *appearance* based. The former methods extract object contours and match them to pre-computed people models. Gavrila et al. [7] proposed the Chamfer System, while Wu and Nevatia [26] propose to use edgelets (small Chamfer segments up to 12 pixels long), in combination with AdaBoost learning, for matching.

Methods based on appearance can be divided into holistic and part-based. **Holistic** approaches model the person as a unique region. Viola and Jones [10] embedded the Haar-like features in a cascaded AdaBoost framework. Dalal and Triggs [4] developed HOG features, which when trained with linear SVM, generate a model used to detect people in a sliding window multi-scale approach. Accepted as one of the most powerful feature descriptors, [17, 28, 18] proposed ways to speed-up the computation of HOGs.

Part-based approaches combine the classification of various parts of the body (head, legs, arms, etc.), instead of classifying the whole person. [14] use Haar wavelets and a quadratic SVM to independently classify four human parts. Wu and Nevatia [27] propose to use the full body, head-shoulder, torso, and legs and three view categories, to train an AdaBoost nested classifier with edgelets as features. Felzenszwalb et al. [6] describe an object detection system based on mixtures of multi-scale deformable part models, having state-of-the-art performance. It utilizes HOGs as features and training with partially labeled data using a latent SVM for data-mining of hard negatives. The final classifier is given by the sum of classifications scores of the ROIs and six different dynamic parts. Similarly, Dollar et al. [5] use a part-based scheme called Multiple Component Learning with Haar features.

Belongie et al. [3] use a Hessian-Laplace keypoint detector and construct a codebook by computing shape context descriptors for each keypoint and clustering them, while

Leibe et al. [11] and Seeman et al. [20] make use of the Implisit Shape Model (ISM).

Holistic methods have a lower complexity than part-based models, however, they do not support partial occlusions or pose variations that are captured by the latter. Regardless of the approach taken, HOG and Shape Contexts seem to be the best feature descriptors independent of the learning method. Hence, in this work we use HOG as feature descriptors and scene specific training (with virtual humans) to capture occlusions and pose variations.

Advances in computer graphics make it possible for very realistic models to be generated on real backgrounds, enriching existing training datasets. Shotton et al. [21] predict 3D positions of body joints from a single depth image extracted from a Kinect camera, and generate the training examples using only synthetic data acquired from MOCAP sequences. In Pischulin et al. [16], the real human datasets are enriched by synthetic samples generated from SCAPE [2]. Lastly, Marin et al. [13] propose a method for pedestrian detection based on training sequences in virtual scenarios where appearance-based pedestrian classifiers are learnt using HOG and linear SVM, providing very good results. Similarly, we train classifiers solely based on virtual humans placed in a 3D model of the scene.

3. 3D Scene Model and Virtual Humans

The focus of the paper is the advantages of viewpoint specific classifiers, rather than the overall machinery including the generation of training images using virtual humans. But this section is provided to discuss and motivate the whole framework.

- *3D Scene Model:* The 3D scene model in this work was obtained by manually measuring a restaurant, and artist creation of a 3D model. CAD models for tables and chairs are readily available. Texture from the real camera image was mapped to the created 3D model.

- *Calibrated camera:* The real camera in the physical scene is calibrated for intrinsic/ extrinsic parameters, which are used to set a virtual camera in the 3D model.

- *Scene Tiling and Associated Classifiers:* The ground plane of the 3D scene model is divided into overlapping tiles (see Section 4.1), which are marked according to their associated human activity - upright, sitting, upright/sitting, or inaccessible. The label will determine the training images for the classifier at that tile.

- *Virtual Humans and Training Images:* A large selection of customizable virtual humans is freely available. Using Maya scripts, a wide range of training images for each tile are obtained by positioning the virtual human on the tile. We experimented with various human heights, weights, genders, and motions (e.g. walking, arm waving etc.). Similarly for the seated case, a series of virtual humans are automatically seated in a 3D chair model and perform a variety of arm motions to provide the training images.

- *Handling Occlusion and Image Edge Effects:* Occlusion by scene infrastructure is correctly handled in the synthetic training images by using the 3D model (e.g. Fig. 1 (red)). Handling occlusion between multiple humans is outside our scope. Similarly some scene tiles will be associated with partial views of humans (Fig. 1 (green)). For such cases, training images are generated for the partial views, thereby enabling detection of partially-truncated humans at image edges. This extends the detection coverage in the live system.

- *Handling Scene Modification:* Minor changes of table and chair position are handled as per Section 4.6. Large-scale changes to the physical layout of the scene would require retraining of the system.

The requirement for a 3D model of the scene is a significant prerequisite for the viewpoint specific approach. But 3D capture continues to demonstrate impressive advances, recent examples including KinectFusion, the Panasonic D-Imager outdoor-operation depth camera, the Occipital Structure Sensor, Zebedee, and Project Tango[2]. State-of-the-art algorithms for 3D reconstruction include [9] and [12]. We believe it is feasible to envisage 3D scans of target scenes as a practical component technology in real-world applications.

4. Method

This section describes the system - firstly the 3D scene model and virtual humans; then the training of classifiers, the run-time people detection for upright humans, and the extension to seated humans; finally the handling of changes in the scene model. Fig. 2 is a schematic overview of the system operation; a summary is shown in Algorithm 1.

4.1. Scene Tiling

Fig. 3a shows a CAD model of a restaurant from the same viewpoint as the real camera in the real scene and Fig. 3b the respective top view, with the ground plane divided into a grid of overlapping square tiles. Tile size was selected to correspond to occupancy by a single average human, while the amount of tile overlap determines the balance between the density of the classifier coverage and computational expense. We used an overlap of 50 percent.

[2] research.microsoft.com/en-us/projects/surfacerecon, www2.panasonic.biz/es/densetsu/device/3DImageSensor/en, structure.io, wiki.csiro.au/display/ASL/Zebedee, www.google.com/atap/projecttango

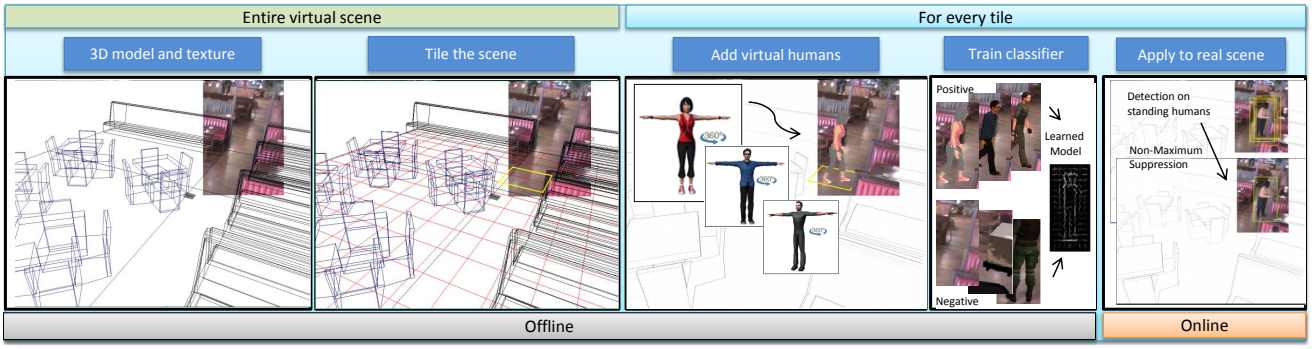


Figure 2. System operation. The starting point is a 3D model of the real scene. The method is (a) divide the ground plane into overlapping tiles, (b) for each tile, position a series of virtual humans on the tile to generate positive training images, and use non-human targets to generate negative training images, (c) train a classifier, (d) at run-time, run all classifiers to perform people detection at the tiles.

Note that tiling also happens outside the camera's field-of-view. A tile that is immediately below the camera and outside of its field-of-view, for example, can still be associated with a visible head and torso for a human located on that tile. Such truncated views of humans will occur for all four sides of the image given an elevated camera that is looking down into a scene. Training of classifiers for tiles that are associated with partial views allows us to significantly extend the coverage of detection and tracking.

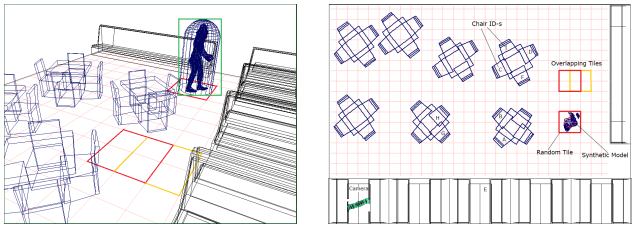
Algorithm 1: System operation.

1 Divide the ground plane into a rectilinear grid of overlapping square tiles T_i;
2 **for** *each* T_i **do**
3 Compute a corresponding area t_i on the camera image plane, suitable to bound a typical human located at T_i;
4 Place a virtual human on T_i, render the image area t_i, and store it as a positive training example;
5 Repeat for virtual humans of different gender, shape, and pose to generate a set of positive training examples;
6 Repeat the above using non-human virtual objects to create a set of negative training examples;
7 Train a classifier C_i for tile T_i;
8 At run-time, run each classifier C_i on its corresponding t_i;

4.2. Generating Training Images using Virtual Humans

Virtual humans are readily available in the graphics community. We used ten male and female models[3] of various sizes and textures, with examples shown in Fig. 4. Each model was registered to Maya's human skeleton to enable animation such as walking or sitting.

[3] from http://www.mixamo.com/



(a) camera view (b) top view

Figure 3. CAD model for a restaurant scene from (a) the same viewpoint as the real camera in the physical scene and (b) a top view. Also shown are the division of the ground plane into overlapping tiles and the capped cylinder surrounding the human figure - the bounding box of the cylinder (shown in green) specifies the image window corresponding to the classifier for that tile.

Figure 4. Virtual Humans. A wide range of models are available, including models parameterized on height, weight, etc.

Training data is generated by iterating through each of the overlapping tiles T_i in the scene. The process carried out at each tile is as follows -

- Position a capped cylinder on the tile as shown in Fig. 3a. The capped cylinder is projected onto the camera image plane, and the bounding box of the projection defines the image rectangle t_i which will be used for all training examples associated with tile T_i[4].
- Position a virtual human on the tile, and render as shown in Fig. 5a. Some tiles may generate partial views, for example due to truncation by the left side

[4] It might seem that a rectangular parallelepiped is a simpler choice than a capped cylinder. However, our experience was that, for tiles closest to the camera viewpoint, the rectangular parallelepiped is an overestimate for a suitable t_i.

Figure 5. **(a)** Virtual human, **(b)** A tile close to the image boundary will generate truncated views, but such tiles are still valuable for human detection and are utilized, **(c)-(d)** Standing training examples, **(e)-(f)** Sitting training examples.

Figure 6. Examples of negative training images.

of the image as shown in Fig. 5b. All tiles, whether associated with full or partial views, are treated in the same way.

- Superimpose the rendered virtual human on the real background image as shown in Fig. 5c-d. Store the image as a positive training example.

- Repeat the previous two steps for a variety of virtual humans and poses of interest for that tile location. We render for each tile a total of 792 different images - 9 human models x 8 orientations at 45^o intervals x 11 articulations. The 11 articulations include walking, waving and clapping poses.

Occlusion at some tiles due to objects in the 3D scene is correctly handled in the rendering as shown for the seated virtual humans in Fig. 5e-f.

For each tile, we also generate 7000 negative training examples of the same window size as the positive set. The negative examples are created using 3D models of non-human objects, using the empty background, and using individual body parts of virtual humans. The latter helps to prevent false positives triggered by body parts. Fig. 6 shows examples of negative training data.

4.3. Training a Classifier per Tile

The training process for an individual tile involves extracting HOG features from the generated training data, and training with a linear SVM. While Dalal et al. [4] train a general human classifier over a 64×128 window, we train specific human classifiers for every tile in the scene. Because the tile specific bounding box comes in different sizes for different tiles, sometimes exceeding the aforementioned window size, we rescale the windows when needed such that the amount of features extracted does not exceed 3780. The latter is the number of features extracted from a window of 64×128 pixels, setting the HOG cell size to 8×8 pixels, the block size to 2×2 cells, the block stride to 8 pixels and the number of gradient bins per cell to 9. We

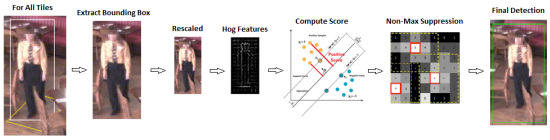


Figure 7. Detection scheme.

use the same HOG settings and we observe no loss in accuracy from rescaling. An improvement in speed both during training and detection is noticed as compared to the non-rescaled bounding boxes. Some bounding boxes need to be padded with extra pixels from the background before rescaling, in order to preserve the width and height constraints of multiplicity with the cell size. The extracted features from the positive and negative rescaled window patches are fed, along with their respective classes, to a linear SVM for training as in [4]. We only perform one round of training without mining the hard negatives. We also probed the effect of training while varying the number of virtual human models used to generate the positive training data, and we noted that increasing the number of models beyond ten did not improve the results.

4.4. People Detection

Having pre-computed the classifiers, we utilize a per tile single-scale approach to detect humans in images from a video stream. For a specific image, we extract from every tile the respective window patch delineated by the pre-computed bounding box, and we rescale it applying the same method explained in Sec. 4.3. The classifiers are applied then to the features extracted from every patch, and a decision whether the patch contains a human or not is obtained, along with the confidence/score of that decision. The firing patches are further filtered based on the scores using a non-maximum suppression scheme, as shown in Fig. 7.

Background Subtraction. Instead of running the classifiers directly on the image, we leverage from our static camera setup and apply a background subtraction scheme for pre-processing. (*Note:* background subtraction is not used in the experimental evaluation, only in our final system). The scheme is based on learning adaptive mixture models of the background scene while incorporating a detector for moving shadows [29, 30]. For each 2D bounding box, the percentage of enclosed foreground pixels over the box size is calculated, and if it is bigger than a threshold, the tile specific classifier is run on top of it. By varying the threshold one can trade off precision with recall, however, at this stage we focus mainly on recall by setting the threshold very low (20%) and letting the classifiers themselves decide whether a human is present or not. The purpose of background subtraction is to act as an unstrict pre-filter to prevent processing in areas with no activity, but without the

need for critical tuning.

Per Tile Confidence Computation. As a raw tile score represents the distance from the hyperplane for that tile, it cannot be used as a comparison over all tiles in the non-maximum suppression scheme below (since every hyperplane differs). Hence the score for a tile is standardized as in Algorithm 2. In this way the scores become comparable.

Algorithm 2: Tile score computation.

1 Let S_c be the current score;
2 Let S_m be the median of all scores computed from applying the tile specific classifier to all the positive training images;
3 Standardized tile score S_t

$$S_t = \frac{S_c}{S_m} \tag{1}$$

Non Maximum Suppression. After running the classifiers on the remaining boxes from the background subtraction step, some false detections in the vicinity of the human subjects might remain. In order to suppress those, we use a standard non maximum suppression scheme which consists of the rejection of lower confidence bounding boxes that have an intersection over the union (IOU) with the others greater than 0.5.

4.5. Standing vs Sitting

The preceding sections present stages from generation of training data to detection of upright humans. This section reviews the modifications made to detect sitting humans. Detection of sitting is a key goal in our application, and we train specifically for it (use of virtual humans makes it readily possible to train for varied types of human pose). Whereas training for standing people happens at every accessible tile, training for sitting people is done only where there are seats, as defined by manual labeling of scene tiles at setup time (see Section 3). An appropriate automatically generated animation of the virtual humans is used to generate sitting pose. This is followed by generation of training imagery, and training of classifiers as already described. The positive training data consists of 720 image patches around the sitting objects (9 virtual models $\times$ 40 articulations (sampled from a sitting, bending and waving animation) $\times$ 2 elevations above the chair), as can be seen in Fig. 5e-f. The negative data are extracted in the same way as described earlier in preceding sections. Only the body orientation corresponding to the specific orientation of the chair or couch where the subject would be sitting is used. The training of the classifiers, confidence computation and detection is performed exactly as for standing people, except that IOU non-maximum suppression is not performed because close proximity of seated humans does not occur in our scenario. In the cases where a standing and sitting detection coincide in the same 3D scene location, the detection with the lowest confidence is deleted. Similarly, it can also happen that standing detections are triggered from sitting people, because of similar gradients around the head and shoulders. We detect such cases by looking for overlap of standing detections and sitting detections in the image space, and avoid false positives by deleting the least confident of the two overlapping boxes from either configuration.

Details	Number
Video frames evaluated	1430
Annotations of standing adults	2415
Annotations of sitting people	5880
Annotations of standing children	50

Table 1. Key figures of our dataset.

4.6. Changes in Scene Model

Minor changes in table and seat location can be treated as a translation i.e. by iterating the classifier across a small search window on the image plane centered on the associated tile. For a significant reconfiguration of the environment, we would retrain.

5. Results

This section describes the experimental results. The system was tested using five hours of imagery from a restaurant[5]. The imagery shows people entering or leaving the scene alone and in groups, and seated diners. For evaluation, we subsampled 1430 images from a total of 8000 images such that no correlation over time occurs, and manually annotated the humans for ground truth, separating them into three categories as presented in Table 1. The bounding boxes were annotated following the same annotation approach used for HOG training with the INRIA dataset [4].

Comparative Analysis. We compare our approach with three methods and use the following terminology -

- *Tile Specific HOG:* refers to our method.
- *Default HOG:* refers to the generic HOG human detector trained on the INRIA dataset [4]. The INRIA dataset consists of generic humans in generic settings, so we make a second comparison with a HOG classifier trained using the annotated dataset from our scene. (*Note:* for this training, we ensured that no training examples were taken from truncated humans at image boundaries). Using two-fold cross validation on three

[5]For confidentiality reasons, full images of the restaurant are not shown and images in the paper are cropped to individual humans or small groups. The camera captured an area of about 10m-by-10m with four tables plus side booths, seating about 40 people, as shown in Fig. 3.

Figure 8. Detections of humans on the real dataset. (a) close by, (b) multiple standing/sitting, (c) truncated/occluded

different set splits, we follow the exact same steps as for the Default HOG, utilizing the same number of positive to negative, window patches of 64×128 pixels and one round of hard negatives mining.

- *Latent SVM:* refers to the state-of-the-art method based on discriminatively trained part based models in [6].

Because Default HOG is trained on whole human bodies, the comparison is performed only for standing/upright detections. For our own method only, we additionally evaluate the gain in accuracy by incorporating sitting detection.

Quantitative Results. As an evaluation metric we use the P-R curves. A true positive is considered any detected box that has an IOU with any of the ground truth boxes greater than 0.5, allowing only one-to-one assignments. For all the methods that we compare to, we find the best scaling factor of the detected boxes that maximizes their performance. As a reminder, the Default HOG uses a sliding window multi-scale scheme over the entire image, as compared to our tile specific single scale scheme. For this reason, we further prune detections that occur in image boundaries as well as at physically impossible positions (here detections on the image top). Similarly, no background subtraction is used for the evaluation step. In Fig. 9(a) , we compare the evaluation results of our method to the Default HOG and Latent SVM. In Fig. 9(b) , the comparison is made to the HOG trained on three dataset splits. Our method shows best performance in this test. This is most probably because a separate classifier is being trained at each tile as compared to the general classifier from the other methods, hence we can handle perspective distortion. Additionally, occlusions are specifically taken care of from the incorporation of the 3D model of the scene, unlike the other methods. In Fig. 9(c) , we show the effect of incorporating sitting people detections on the standing people curve, noticing an improvement in precision. This is due to the elimination of false positives resulting from sitting people.

Runtime Comparison. We analyze the time taken by the sliding window multi-scale approaches to find all the detections in an image with resolution of 640×480 pixels, as compared to our per tile single scale approach. The algorithms do not use GPU implementations and run on a Unix OS with 2.55 GHz 2 Quad processor. Using a scale factor

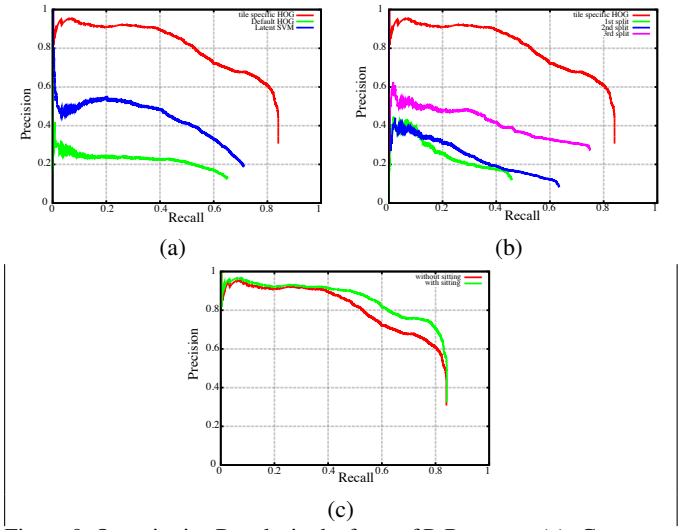


Figure 9. Quantitative Results in the form of P-R curves. (a): Comparison between our method (red), Default HOG (green), and Latent SVM (blue). (b): Comparison between our method (red) and the standard HOG detector trained on our own dataset using three splits. (c): Change in performance when using only our standing people detector (red) versus including also sitting people (green).

of 1.05 for the Default HOG and a single threaded detection for Latent SVM we notice that our method is faster, as can be seen in Table 2. If we compare the number of HOG windows computed per image, our method computes a maximum of 800 windows assuming that nothing is filtered from the background subtraction step. On the other hand, the Default HOG does 24320 computations over 27 scales. This huge performance gap however is narrowed utilizing the properties of the Fourier transforms on sliding window approaches. An increase in performance for all the methods would occur if GPU or multi-threaded implementations are used.

6. Conclusion

This paper describes viewpoint specific classifiers. Training data is obtained by creating a 3D model of the target scene and inserting virtual humans of varied shape, animated with physically correct behavior, to generate training imagery. The resulting classifiers are applied successfully

Algorithm	Time [s]
Tile Specific HOG	0.35
Default HOG	0.67
Latent SVM	5

Table 2. Runtime comparison of the different algorithms. Our method is Tile Specific HOG.

to real imagery of the scene. The approach was demonstrated on a restaurant scene as an exemplary scenario, however it could equally well handle other types of scene of the same complexity. The contributions of the paper are the following. Firstly, the description of a complete system for people detection based on the viewpoint specific concept, intended to be practical for real-world deployment. Secondly, demonstration of people detection for standing and sitting humans in a complex scene with occluders. Thirdly, a comparative analysis of the system with state-of-the-art algorithms, to demonstrate benefits of the approach.

7. Acknowledgements

We would like to thank Maurizio Nitti for the help with the 3D scene and human models as well as Marcin Eichner for his valuable inputs.

References

[1] B. Alexe, N. Heess, Y. W. Teh, and V. Ferrari. Searching for objects driven by context. In *NIPS*, 2012.

[2] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. SCAPE: shape completion and animation of people. *ACM TOG*, 2005.

[3] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 2002.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[5] P. Dollàr, B. Babenko, S. Belongie, P. Perona, and Z. Tu. Multiple component learning for object detection. In *ECCV*, 2008.

[6] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 2010.

[7] D. M. Gavrila, J. Giebel, and S. Munder. Vision-based pedestrian detection: The PROTECTOR system. In *IV*, 2004.

[8] H. Grabner, P. M. Roth, and H. Bischof. Is pedestrian detection really a hard task? In *PETS*, 2007.

[9] M. Habbecke and L. Kobbelt. Laser brush: a flexible device for 3D reconstruction of indoor scenes. In *SPM*, 2008.

[10] M. Jones, P. Viola, P. Viola, M. J. Jones, D. Snow, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *ICCV*, 2003.

[11] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *IJCV*, 2008.

[12] Lu-Xingchang and Liu-Xianlin. Reconstruction of 3D model based on laser scanning. In *3D-GIS*, 2006.

[13] J. Marn, D. Vàzquez, D. Gerònimo, and A. M. Lòpez. Learning appearance in virtual scenarios for pedestrian detection. In *CVPR*, 2010.

[14] A. Mohan, C. Papageorgiou, and T. Poggio. Example-based object detection in images by components. *PAMI*, 2001.

[15] M. Munaro and A. Cenedese. Scene specific people detection by simple human interaction. In *ICCV*, 2011.

[16] L. Pishchulin, A. Jain, M. Andriluka, T. Thormaehlen, and B. Schiele. Articulated people detection and pose estimation: Reshaping the future. In *CVPR*, 2012.

[17] F. Porikli. Integral histogram: A fast way to extract histograms in Cartesian spaces. In *CVPR*, 2005.

[18] V. A. Prisacariu and I. Reid. FastHOG- a real-time GPU implementation of HOG. Technical report, University of Oxford, 2009.

[19] P. M. Roth, S. Sternig, H. Grabner, and H. Bischof. Classifier grids for robust adaptive object detection. In *CVPR*, 2009.

[20] E. Seemann, B. Leibe, and B. Schiele. Multi-aspect detection of articulated objects. In *CVPR*, 2006.

[21] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011.

[22] S. Stalder, H.Grabner, and L. V. Gool. Exploring context to learn scene specific object detectors. In *PETS*, 2009.

[23] S. Sternig, P. M. Roth, and H. Bischof. Learning of scene-specific object detectors by classifier co-grids. In *AVSS*, 2010.

[24] P. Sudowe and B. Leibe. Efficient use of geometric constraints for sliding-window object detection in video. In *ICVS*, 2011.

[25] M. Wang, W. Li, and X. Wang. Transferring a generic pedestrian detector towards specific scenes. In *CVPR*, 2012.

[26] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by Bayesian combination of edgelet part detectors. In *ICCV*, 2005.

[27] B. Wu and R. Nevatia. Detection and tracking of multiple, partially occluded humans by Bayesian combination of edgelet based part detectors. *IJCV*, 2007.

[28] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *CVPR*, 2006.

[29] Z. Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *ICPR*, 2004.

[30] Z. Zivkovic and F. van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 2006.