Shape-Based Human Detection and Segmentation via Hierarchical Part-Template Matching

Zhe Lin, *Member*, *IEEE*, and Larry S. Davis, *Fellow*, *IEEE*

Abstract—We propose a shape-based, hierarchical part-template matching approach to simultaneous human detection and segmentation combining local part-based and global shape-template-based schemes. The approach relies on the key idea of matching a part-template tree to images hierarchically to detect humans and estimate their poses. For learning a generic human detector, a pose-adaptive feature computation scheme is developed based on a tree matching approach. Instead of traditional concatenation-style image location-based feature encoding, we extract features adaptively in the context of human poses and train a kernel-SVM classifier to separate human/nonhuman patterns. Specifically, the features are collected in the local context of poses by tracing around the estimated shape boundaries. We also introduce an approach to multiple occluded human detection and segmentation based on an iterative occlusion compensation scheme. The output of our learned generic human detector can be used as an initial set of human hypotheses for the iterative optimization. We evaluate our approaches on three public pedestrian data sets (INRIA, MIT-CBCL, and USC-B) and two crowded sequences from Caviar Benchmark and Munich Airport data sets.

Index Terms—Generic human detector, part-template tree, hierarchical part-template matching, pose-adaptive descriptor, occlusion analysis.

1 Introduction

Human detection is a fundamental problem in video surveillance. It can provide an initialization for human segmentation. More importantly, robust human tracking and identification are highly dependent on reliable detection and segmentation in each frame since better segmentation can be used to estimate more accurate and discriminative appearance models. Although the problem of human detection has been well studied in vision, it still remains challenging due to highly articulated body postures, viewpoint changes, varying illumination conditions, occlusion, and background clutter. Combinations of these factors result in large variability of human shapes and appearances in images. Our goal is to develop a robust and efficient approach to detect and segment (possibly partially occluded) humans under varying poses and camera viewpoints.

1.1 Previous Work

Many approaches have been introduced for human detection over the last decade and significant progress has been made in terms of robustness and efficiency.

Most commonly, human detection is formulated as a binary sliding window classification problem, i.e., a fixed-size window is scanned over an image pyramid and bounding boxes are localized around humans based on some nonmaximum suppression procedure. In terms of visual cues, shape has been the most dominant cue for detecting humans in still images due to large appearance variability. Motion has been widely used as a complimentary cue for detecting humans in videos.

In the object category detection literature, shape modeling or shape feature extraction schemes can be roughly classified into two categories. The first category models human shapes globally or densely over image locations, e.g., an overcomplete set of Haar wavelet features in [1], rectangular features in [2], histograms of oriented gradients (HOGs) in [3], locally deformable Markov models in [4], or covariance descriptors in [5]. Global, dense feature-based approaches such as [3], [5] are designed to tolerate some degree of occlusions and shape articulations with a large number of samples and have been demonstrated to achieve excellent performance with well-aligned, more or less fully visible training data. In [6], [7], [8], human shapes are more directly modeled as a set of global shape templates organized in a hierarchical tree. Due to the nature of global modeling, these approaches require a large number of positive training data.

The second category models an object shape using sparse local features or as a collection of visual parts. Local feature-based approaches learn body part and/or full-body detectors based on sparse interest points and descriptors [9], [10], [11], from predefined pools of local curve segments [12], [13], a contour segment network [14], k-adjacent segments [15], or edgelets [16]. Part-based approaches model an object shape as a rigid or deformable configuration of visual parts [9], [17], [16], [18]. Part-based representations have been shown

- *Z. Lin is with the Advanced Technology Labs, Adobe Systems Incorporated, 345 Park Avenue, San Jose, CA 95110. E-mail: zlin@adobe.com.*
- *L.S. Davis is with the Department of Computer Science, University of Maryland, 3301 A.V. Williams Building, College Park, MD 20742. E-mail: lsd@cs.umd.edu.*

Manuscript received 16 Oct. 2008; revised 2 July 2009; accepted 25 Nov. 2009; published online 22 Dec. 2009.
Recommended for acceptance by A. Srivastava, J.N. Damon, I.L. Dryden, and I.H. Jermyn.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMISI-2008-10-0709.
Digital Object Identifier no. 10.1109/TPAMI.2009.204.

0162-8828/10/$26.00 © 2010 IEEE Published by the IEEE Computer Society

to be very effective for handling partial occlusions. In [17], several part detectors are trained separately for each body part and combined with a second-level classifier. Mikolajczyk et al. [9] use local features for part detection and assemble the part detections probabilistically. Wu and Nevatia [16] introduce edgelet features for human detection. They extend this approach to a general object detection and segmentation approach by designing local shape-based classifiers [19]. Shet et al. [18] propose a logical reasoning-based method for efficiently assembling part detections. One problem with these part-based detection approaches is that, in very cluttered images, too many detection hypotheses may be generated and a robust assembly method (e.g., boosting) is thus needed to combine these detections.

Deformable part or patch-based approaches such as [20], [21], [22], [23] established an elegant, probabilistic framework for modeling object shape articulation. They have been very successful for many vision applications such as handwritten digit recognition, face detection, and object categorization, etc. These approaches commonly represent an object as a collection of parts and enforce certain constraints on spatial configuration of parts to handle object shape articulation. For example, Amit and Trouve [23] proposed a nice, part-based object recognition framework (called patchwork of parts), which is capable of performing object detection and handling occlusion precisely. They applied it successfully to handwritten digit recognition and face detection. Recently, Felzenszwalb et al. [24] combined a deformable part-based model with support vector machines (SVMs) for detecting highly articulated objects in challenging real-world images. This approach achieved very promising results on many object categories in the PASCAL challenges.

Global approaches require training a single, strong classifier; hence, they are much simpler than part-based approaches, where each part needs to be trained separately and an additional assembly classifier is must also be trained. On the other hand, part (or local feature)-based approaches can more readily handle partial occlusions and are more flexible in dealing with shape articulations compared to global approaches. The shape modeling schemes can also be combined with appearance cues for simultaneous detection and segmentation [7], [25], [26]. Shape cues are also combined with motion cues for human detection in [27], [28], simultaneous detection, and segmentation in [29].

From a learning perspective, there are generative and discriminative approaches for learning object detectors. Some generative approaches construct a tree-based data structure for efficient shape matching [6], [7], [8]. Typically, a score is computed for each detection window based on template matching and nearest neighbor search. Most commonly used discriminative approaches include cascaded adaboost classifiers [2], [5], [27], [30] and support vector machines [3], [28], [31]. Dalal and Triggs [3] introduced HOG features and provided an extensive experimental evaluation using linear and gaussian-kernel SVMs as the test classifiers. Later, Zhu et al. [30] improved its computational efficiency significantly by utilizing a boosted cascade of rejectors. Recently, Tuzel et al. [5] reported significantly better detection performance than [3] on the INRIA data set. They use covariant matrices as image descriptors and classify patterns on Riemannian manifolds. Similarly, Maji et al. [31] also demonstrate promising results using multilevel HOG descriptors and faster (histogram intersection) kernel SVM classification. In [32], twofold adaboost classifiers are adopted for simultaneous part selection and pedestrian classification. Wu and Nevatia [33] combine different features in a single classification framework. Pang et al. [34] introduced a multiple instance learning-based scheme for better aligning positive examples in training images.

For surveillance scenarios, motion blob information provides very reliable cues for human detection. Numerous approaches have been introduced for detecting and tracking humans using motion blob information. These blob-based approaches are computationally more efficient than purely shape-based generic approaches, but have a common problem that the results crucially depend on background subtraction or motion segmentation. These approaches are mostly developed for detecting and tracking humans under occlusion. Some earlier methods [35], [36] model the human tracking problem by a multiblob observation likelihood given a human configuration. Zhao and Nevatia [37] introduce an MCMC-based optimization approach to human segmentation from foreground blobs. They detect heads by analyzing edges surrounding binary foreground blobs, formulate the segmentation problem in a Bayesian framework, and optimize by modeling jump and diffusion dynamics in MCMC to traverse the complex solution space. Following this work, Smith et al. [38] propose a similar transdimensional MCMC model to track multiple humans using particle filters. Later, an EM-based approach was proposed by Rittscher et al. [39] for foreground blob segmentation. Zhao et al. [40] use a part-based human body model to fit binary blobs and track humans.

Our motivations are summarized as follows: First, hierarchical template matching [6], [8] is a convenient way to efficiently integrate detection and segmentation of shapes, but it is computationally expensive due to the necessity of collecting and matching with a large number of global shape templates. Next, previous discriminative approaches such as [3], [5], [33] mostly train a binary classifier on a large number of positive and negative samples where humans are roughly center-aligned. These approaches represent appearances by concatenating information along 2D image coordinates for capturing spatially recurring local shape events in training data. However, due to highly articulated human poses and varying viewing angles, a very large number of (well-aligned) training samples are required; moreover, the inclusion of information from whole images inevitably makes them sensitive to biases in training data (in the worst case, significant negative effects can occur from arbitrary image regions); consequently, the generalization capability of the trained classifier can be compromised. Finally, although recent deformable part-based models such as [24] and multiple instance-based learning schemes such as [41] are very effective for localizing objects, they are limited in estimating more precise shape and pose segmentations from the detection process due to simple rectangular part-shape assumptions.

1.2 Overview of Our Approach

In this paper, we tackle the difficult problem of simultaneously detecting and segmenting multiple (possibly partially occluded) humans. For achieving the goal, we

TABLE 1
A Table of Notations

notation	definition or explanation
$\theta_j, j = ht, ul, ll$	The parameters of each part model (head torso, upper legs, and lower legs).
$L_0, L_{1,i}, L_{2,i,j}, L_{3,i,j,k}$	Individual nodes (indices) of the part-template tree.
K	The number of nodes (part models) in the first layer.
$\|G\|, O$	Edge gradient magnitude and orientation map.
T	A part-template characterized by its boundary sample points.
f_T or $f(T)$	A part-template matching score to image (gradient) observation.
$F_{\mathbf{w}}$	A score function for a part-combination matching score by weight $\mathbf{w}$.
$D_m, m = 1, 2...M$	M part detectors formed by different weight vectors $\mathbf{w}_m$.
$\mathbf{u} = \{u_1, u_2, ...u_N\}$	$\mathbf{u}$ represents an unordered set of initial hypotheses.
$L(u_i)$	Full-body matching score of the hypothesis u_i before occlusion compensation.
$L(u_i\|I_{occ})$	Full body matching score after occlusion compensation.
Φ	Objective function for multiple occluded hypotheses optimization.

propose a hierarchical part-template matching approach [42] and combine it with discriminative learning for building a generic human detector. Given an input still image, the detector returns human bounding boxes as well as precise shape segmentation masks. Our approach takes advantages of both local part-based and global template-based human detectors by decomposing global shape models and constructing a part-template tree to model human shapes flexibly and efficiently. Shape observations (edges or local gradient orientations) are matched to the part-template tree efficiently to determine a reliable set of human detection hypotheses. Shapes and poses are estimated automatically through synthesis of part detections. Our approach is similar to [24], [41], [43] in the spirit of allowing part deformations, but our model detects and segments humans simultaneously and explicitly handles partial occlusions.

Using the hierarchical part-template matching scheme, we extract features adaptively in the local context of poses, i.e., we propose a pose-adaptive feature extraction method [44] for better discriminating humans from nonhumans. The intuition is that pose-adaptive features produce much better spatial repeatability of local shape events. Specifically, we segment human poses on both positive and negative samples[1] and extract features adaptively in local neighborhoods of pose contours, i.e., in the pose context. The set of all possible pose instances is mapped to a canonical pose such that points on an arbitrary pose contour have one-to-one correspondences to points in the canonical pose. This ensures that our extracted feature descriptors correspond well to each other, and are also invariant to varying poses.

For extending the tree matching algorithm to multiple occluded humans, a set of detection hypotheses is estimated by our generic human detector and is iteratively refined/optimized through matching score reevaluation and fine occlusion analysis. For meeting the requirement of real-time surveillance systems, we also combined the approach with background subtraction to improve efficiency, where region information provided by foreground blobs is combined with shape information from the original image for improving robustness. Results show that our approach achieves state-of-the-art accuracy on several detection benchmarks, and the segmentation results are very good. Our approach is robust to noisy background subtractions or motion segmentations in detecting multiple humans from videos. Our extended approach outputs a set of human bounding boxes, occlusion ordering, as well as precise shape segmentations given a static video sequence.

Our main contributions are summarized as follows:

- A part-template tree model and its automatic learning algorithm are introduced for simultaneous human detection and pose segmentation. The approach combines popular local part-based object detectors with global shape-template-based schemes.
- A fast hierarchical part-template matching algorithm is proposed to estimate human shapes and poses by matching local image cues such as gradient magnitudes and/or orientations. Human shapes and poses are represented by part-based parametric models.
- Estimated optimal poses are used to impose spatial priors (for possible humans) for encoding pose-adaptive features in the local pose contexts. One-to-one correspondence is established between sets of contour points of an arbitrary pose and a canonical pose.
- The tree matching algorithm is also extended to handle multiple occluded human detection and segmentation in challenging surveillance scenarios. We estimate human configuration by performing optimization in a greedy fashion based on an iterative process of shape matching score reevaluation and fine occlusion analysis.

For better understanding of our approaches in subsequent sections, we summarized notations in Table 1.

The remainder of the paper is organized as follows: Section 2 introduces the details of our hierarchical part-template matching approach. Section 3 describes the pose-adaptive feature extraction method for learning generic human detectors. Section 4 addresses our extended approach to multiple human detection and segmentation. Section 5 briefly introduces the incorporation of motion and

1. For negative samples, pose estimation is forced to proceed even though no person is in them.

geometry cues to the framework. Section 6 presents experiments and evaluations. Finally, Section 7 concludes the paper and discusses possible future extensions.

2 Hierarchical Part-Template Matching

We introduce a hierarchical shape matching approach to efficiently search for rough human poses (shapes) and compute a matching score for each candidate detection window. For achieving this, we take advantage of local part-based and global shape-template-based approaches by combining them in a unified top-down and bottom-up search scheme. Specifically, we extend the hierarchical template matching method in [6], [8] by decomposing the global shape models into parts and constructing a new part template-based tree that captures appearance correlations between part models from the training database of human shapes.

2.1 Generating the Part-Template Tree Model

Due to the high degree of freedom of human pose space, learning precise part-based human pose/shape models directly from real images would require a very large ground truth silhouette data set and obtaining aligned parts for all training examples is very challenging. We alleviate this problem by a two-phase pose learning scheme, which works well even without any ground truth silhouettes.[2] We first generate a flexible set of global shape models by part synthesis using a simple pose generator and then construct a part-template hierarchy using a body part decomposer. Parallelogram-shaped parts are spatially combined for modeling rough human part shapes. We obtained part-template models by decomposing the global shape models spatially and construct a part-template tree to capture human pose variation. Next, we refine the synthesized tree model by learning from a small set of real pose images.

2.1.1 Synthesizing Global Shape Models

For capturing articulation of human poses, we represent the human body with six part regions—head, torso, pair of upper legs, and pair of lower legs. Global shape models are synthesized by combining these part regions spatially, i.e., a union of these part region instances. The shape of each part region is modeled by a horizontal parallelogram characterized by its center location (2 dofs), size (2 dofs), and orientation parameters (1 dof). Thus, the number of degrees of freedom for the whole body is ideally $5 \times 6 = 30$. We generate these global models mainly in order to obtain an automatically pose-aligned compact set of part template models (characterized by both edge and region information) by spatial decomposition.

For simplifying the initial tree construction, global shapes are synthesized using only six degrees of freedom (head position, torso width, orientations of upper/lower legs) given the torso position as the reference, and the remaining parameters (or dofs) are regarded as hidden variables estimated in the online detection/test phase. Heads and torsos are simplified to vertical rectangles (fixed orientations) and the rectangular shapes are further modified to have rounded shapes at corners to make them more realistic. The selected six parameters are evenly quantized into $\{3, 2, 3, 3, 3, 3\}$ values in their ranges,[3] respectively. Denser quantization and larger ranges would improve accuracy but we found that the above quantization and ranges are sufficient for handling standing humans. Finally, instances of part regions are independently combined to form $3 \times 2 \times 3 \times 3 \times 3 \times 3 = 486$ global shape models. Examples of generated global shape models are shown in Fig. 1a.

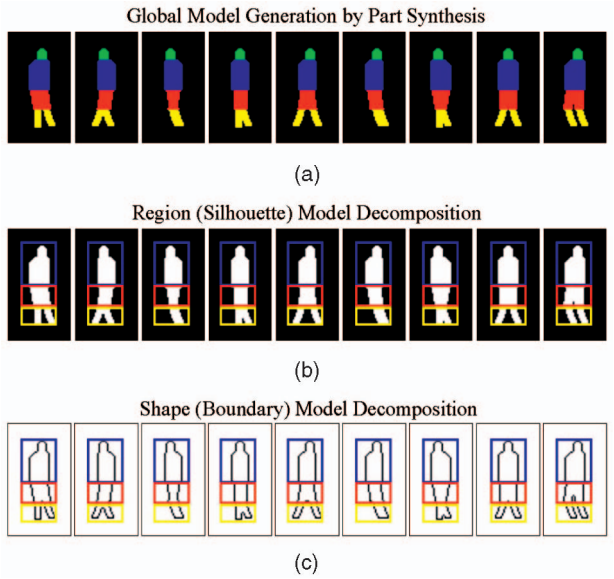


Fig. 1. (a) Generation of global shape models by part synthesis, (b) decomposition of global silhouette and boundary models into region, and (c) shape part-templates.

2.1.2 Generating Parts by Decomposition

In order to obtaining parts from the global shape models, we first binarize the global shape models in Fig. 1a to obtain silhouettes (Fig. 1b) and extract boundaries of the silhouettes (Fig. 1c). Next, silhouettes and boundaries are decomposed into three parts (head-torso (ht), upper legs (ul), and lower legs (ll)) as shown in Figs. 1b and 1c. This will generate a set of part regions and a set of part shapes (curves). By decomposing the global models into local part models, we allow deformation of each part and spatial relations between parts to handle a wider range of human shape articulations. The parameters of the three parts ht, ul, ll are denoted by θ_{ht}, θ_{ul}, and θ_{ll}, where each parameter $\theta_j = (\theta_j^{ind}, \theta_j^{loc})$ consists of the index θ_j^{ind} of the corresponding part (e.g., $\theta_{ul}^{ind} = 3$ means the third part-template) and the location parameter θ_j^{loc} encodes the part's location relative to a detection window. The number of global shape models is determined by the degree of freedom of each part regions (before synthesis), and the number of part-templates is equal to the number of nodes in the tree, so it is much smaller than the global models.

2.1.3 Constructing an Initial Tree Model Using Parts

Given the set of indexed parts, a part-template tree is constructed by placing the decomposed part regions and

2. In this case, we can impose a uniform prior on the branching probability distribution for the tree.

3. The range of each parameter is determined empirically, e.g., the angle of a single leg part is in a range $(-30', 30')$.

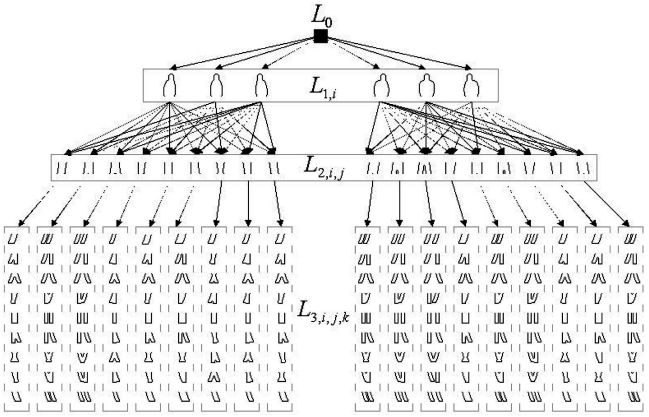


Fig. 2. An illustration of the part-template tree model. The different part-templates are obtained by decomposing the global shape models in a top-town manner. Each part in the tree has a default location/size and is characterized by both shape and region information.

boundary fragments into a tree as illustrated in Fig. 2. The tree edges (or links) are determined automatically from the relation between part indices in the global shape models. The tree has four layers denoted by L_0, L_1, L_2, and L_3, where L_0 is the (empty) root node, L_1 consists of side view head-torso templates $L_{1,i}$, $i = 1, 2, 3$ and front/back view head-torso templates $L_{1,i}$, $i = 4, 5, 6$, and similarly, L_2 and L_3 consist of upper and lower leg poses for side and front/back views. Hence, each part-template in the tree represents an instance of a part, and can also be viewed as a parametric model, where part location and sizes are the model parameters.

The number of part-templates in the tree is directly determined by the number of generated global shape models. A much larger set of part-templates can be obtained by finer discretization and wider ranges of parameters when generating the global shape models but the matching performance only slightly improves with a linear increase in computational requirements w.r.t. the number of templates in the tree. As shown in the figure, the initially generated tree consists of 186 part-templates, i.e., 6 ht models, 18 ul models, 162 ll models, and organized hierarchically based on the layout of human body parts in a top-to-bottom manner. Due to the tree structure, a fast hierarchical shape (or pose) matching scheme can be applied using the model. For example, using hierarchical part-template matching (which is explained later), we only need to match 24 part-templates to account for the complexity of matching 486 global shape models using the method in [6], so it is extremely fast. This initial part-template tree model is visualized in Fig. 2. The initial tree structure and part-templates are constructed by synthetic silhouettes as described above, but it is refined by learning from real images, which is described in the following section.

In Fig. 2, any tree path (from the root node to a leaf node) uniquely determines a global shape model, and a unique tree path is associated with any of the global shape models.

2.2 Learning the Part-Template Tree

The set of all global shape models is formed by enumerating all possible part's (or parallelogram's) parameters, so the constructed part-template tree does not contain any prior statistics from real human silhouettes. So, we learn the priors on the appearance probabilities of part-templates by matching them with real human silhouettes. Since we place the part template models in a tree configuration, the prior is estimated as conditional probability distributions (branching probabilities at each internal tree node).

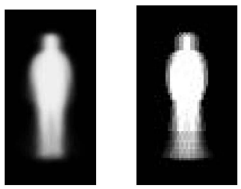
In order to more efficiently and reliably estimate human shapes and poses by matching the tree to an input image, we learn the probabilities associated with the edges of the part-template tree model using real images. Specifically, the learning is performed by matching the tree to a set of human silhouette images. The goal is to explicitly estimate branching probability distributions, i.e., conditional distributions of tree edges (arrows in Fig. 2 connecting nodes between two consecutive layers) in each of the tree layers for handling the observed range of shape articulation. For example, given the empty node at Layer L_0, we estimate the probability distribution of six ht models at Layer L_1 based on their occurrence frequencies in the set of training silhouettes.

The silhouette training set consists of 404 (plus mirrored versions) binary silhouette images (white foreground and black background). Those binary silhouette images were obtained by manually segmenting a subset of positive image patches of the INRIA person database. Each of the training silhouette images is passed through the tree from the root node to a leaf node for identifying the optimal path (corresponding to the largest matching score). The optimal path is estimated using a greedy search algorithm by selecting a locally optimum branch at each node. This can be replaced by a more sophisticated dynamic programming algorithm, but we found that the greedy search works very well and is much faster. The matching score for each node is computed as the degree of coverage between the part-template at the current node and the observed silhouette (i.e., the proportion of pixels inside the hypothesis consistent with the training silhouette). This process is repeated for all the training silhouettes and an optimal path is estimated for every silhouette. Then, based on the set of paths, a branching probability distribution (conditional distribution of branching edges) is estimated for each node based on occurrence frequencies.

Given the learned branching probabilities, any tree path from the root to leaf can be associated with three probability values (for three parts ht, ul, and ll, respectively). Each tree node now carries a binary image of the part-template, its boundary sample point coordinates, and a branching probability distribution obtained from the tree learning step. Since there is a one-to-one correspondence between a path and a global shape model, each global shape model is now represented by a soft coverage map. We accumulate the soft coverage maps for all paths to compute the average global shape for the learned tree. Fig. 3 validates our tree learning approach by showing that the learned average global shape is very similar to the average of all training silhouettes.

2.3 Hierarchical Part-Template Matching

Given a test image, we use a scanning window approach for estimating the optimal pose for each candidate detection window. For each window, we match the learned tree to image observations (edges or edge orientations) to estimate the optimal tree path and associated location parameters for



(a) (b)

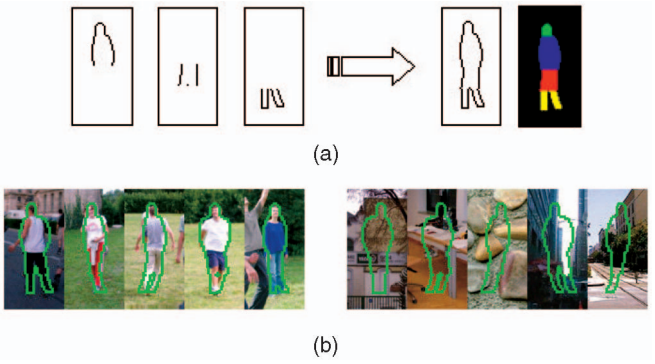
Fig. 3. A comparison of (a) the average of all training silhouettes and (b) the average of the soft probability maps for all tree paths.

the nodes of the path. Similarly to the model used for tree learning, the overall matching score for a particular detection window is simply modeled as a *summation* of matching scores of all nodes along that path. But, differently from the learning, the matching in the test phase is performed on edges or gradient orientations instead of silhouettes, and also the score of each node is computed as the product of the part-template matching score and the prior probability of that node learned in the training phase. We qualitatively verified that incorporating the branching prior probability gives better robustness to pose estimation. Another difference is that, in the test phase, we allow the part-template locations to be shifted locally and an optimal location is estimated for each part-template, as opposed to fixing part-template locations. This is similar to the approaches of Amit and Trouve [23] and Felzenszwalb et al. [24], where each part can be adjusted to its locally optimum location.

We match individual part-templates and compute part-template matching scores using a method similar to Chamfer matching [6]. The matching score of a sample point on the part-template contour is measured by edge orientation matching. The goal is to estimate the optimal human pose (corresponding to a tree path and location estimates of each part-templates on the path), which is most consistent with the image observations.

The optimization problem can be solved by dynamic programming to achieve globally optimal solutions. But, this algorithm is computationally too expensive to densely scan all hypothetical windows. For efficiency, we use a fast K-fold greedy search algorithm. We keep scores for all nodes ($k = 1, 2 \ldots K$) in Layer L_1 instead of estimating the best k and a greedy procedure is individually performed for each of those K nodes (or threads).

Pose model parameters estimated by the hierarchical part-template matching algorithm are directly used for pose segmentation by part synthesis (region connection). Fig. 4 shows the process of global pose (shape) segmentation by the part-template synthesis.



(a)

(b)

Fig. 4. An illustration of shape/pose segmentation. (a) Best part-template estimates (three images on the left side designate a path from L_0 to L_3) are combined to produce the final global shape and pose segmentations (two images on the right side). (b) Example pose (shape) segmentation on positive and negative examples.

3 Pose-Adaptive Descriptors

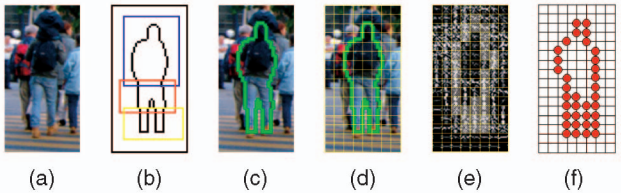
For applying our part-template tree model and hierarchical part-template matching algorithm to discriminative human detection, we introduce a pose-adaptive feature computation method for detecting humans from images using standard machine learning techniques such as SVM and Boosting.

3.1 Overview of the Approach

In our training and testing data sets, positive samples all consist of 128×64 image patches. Negative samples are randomly selected from raw (person-free) images; positive samples are cropped (from annotated images) such that persons are roughly aligned in location and scale. For each training or testing sample, we first compute a set of histograms of (gradient magnitude-weighted) edge orientations for nonoverlapping 8×8 rectangular regions (or cells) evenly distributed over images. Motivated by the success of HOG descriptors [3] for object detection, we employ coarse-spatial and fine-orientation quantization to encode the histograms, and normalization is performed on groups of locally connected cells, i.e., blocks. Then, given the orientation histograms and a candidate detection window, hierarchical part-template matching is performed to estimate the optimal poses. Given the pose and shape estimates, block features closest to each pose contour point are collected; finally, the histograms of the collected blocks are concatenated in the order of pose correspondence to form our feature descriptor. As in [3], each block (consisting of four histograms) is normalized before collecting features to reduce sensitivity to illumination changes. The one-to-one point correspondence from an arbitrary pose model to the canonical one reduces sensitivity of extracted descriptors to pose variations. Fig. 5 shows an illustration of our feature extraction process. Details of dense feature extraction and the tree-based pose inference are described in the following section.

3.2 Low-Level Feature Representation

For pedestrian detection, histograms of oriented gradients (HOGs) [3] exhibited superior performance in separating



(a) (b) (c) (d) (e) (f)

Fig. 5. Overview of our feature extraction method. (a) A training or testing image. (b) Part-template detections. (c) Pose/shape segmentation. (d) Cells overlaid onto pose contours. (e) Orientation histograms and cells overlapping with the pose boundary. (f) Block centers relevant to the descriptor.

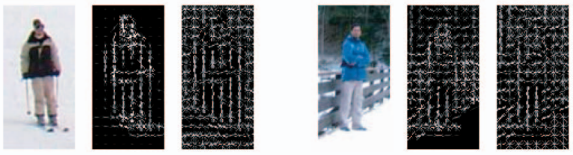


Fig. 6. Examples of two training samples and visualization of corresponding (unnormalized and L_2-normalized) edge orientation histograms.

image patches into human/nonhuman. These descriptors ignore spatial information locally and, hence, are very robust to small alignment errors. We use a very similar representation as our low-level feature description, i.e., (gradient magnitude-weighted) edge orientation histograms.

Given an input image, we calculate gradient magnitudes $|G|$ and edge orientations O using simple difference operators $(-1, 0, 1)$ and $(-1, 0, 1)^t$ in the horizontal x and vertical y directions, respectively. We quantize the image region into local 8×8 nonoverlapping cells, each represented by a histogram of (unsigned) edge orientations (each surrounding pixel contributes a gradient magnitude-weighted vote to the histogram bins). Edge orientations are quantized into $N_b = 9$ orientation bins $[k\frac{\pi}{N_b}, (k+1)\frac{\pi}{N_b})$, where $k = 0, 1 \ldots N_b - 1$. For reducing aliasing and discontinuity effects, we also use trilinear interpolation as in [3] to vote for the gradient magnitudes in both spatial and orientation dimensions. Additionally, each set of neighboring 2×2 cells forms a block. This results in overlapping blocks, where each cell is contained in multiple blocks. For reducing illumination sensitivity, we normalize the group of histograms in each block using L_2 normalization with a small regularization constant ϵ to avoid dividing-by-zero. Fig. 6 shows visualizations of extracted low-level features for two example image patches.

The above computation results in our low-level feature representation consisting of a set of raw (cell) histograms (gradient magnitude-weighted) and a set of normalized block descriptors indexed by image locations. We use both normalized and unnormalized histograms for our descriptor computation algorithm. Unnormalized (raw) histograms carry gradient magnitude information, so they are used for pose matching (matching histograms against edge orientations of the pose contours in the tree), while normalized histograms are used for final descriptors based on the estimated poses.

Note that, in our implementation, all positive and negative training examples are of size 128×64, so, during the training phase, there is no scale variation. All features (edge gradients and orientations) are computed at the same scale at training. During testing, a pyramid of the input image is constructed and a window of the same size 128×64 is scanned to handle scale variation. The (downsampling) scale factor used to construct the image pyramid is set to 1.1.

3.3 Pose Inference on the Low-Level Features

Given a low-level feature representation for a detection window and the learned part-template tree, an optimal tree path is estimated based on the hierarchical part-template matching scheme introduced in the previous section. The part-template score is measured by an average

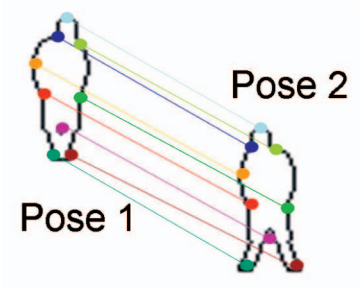


Fig. 7. An illustration of pose alignment by one-to-one contour point correspondence. Only a subset of key contour points is shown here.

of gradient magnitudes of corresponding orientation bins in each block of our low-level feature representation. Here, the score is measured by orientation consistency instead of distances between edges in traditional chamfer matching [6]. The score of each template point is measured using location-based lookup tables for speed. Magnitudes from neighboring histogram bins are weighted to reduce orientation biases and regularize the matching scores of each template point.

More formally, let $O(\mathbf{t})$ denote the edge orientation at contour point $\mathbf{t}$; its corresponding orientation bin index $B(\mathbf{t})$ is computed as: $B(\mathbf{t}) = [O(\mathbf{t})/(\pi/9)]$ (where $[x]$ denotes the maximum integer less than or equal to x). For each contour point $\mathbf{t}$, we first identify the closest 8×8 cell in the detection window. Let $H = \{h_i\}$ (where i denotes the histogram bin index) be the (unnormalized) orientation histogram at the contour location $\mathbf{t}$, the matching score $f(\mathbf{t})$ at $\mathbf{t}$ is computed as

$$f(\mathbf{t}) = \sum_{b=-\delta}^{\delta} w(b) h_{B(\mathbf{t})+b}, \tag{1}$$

where δ is a neighborhood range and $w(b)$ is a symmetric weight distribution.[4] Given a part-template T (a collection of boundary sample points on the part-template), the matching score f_T of the part-template is computed as the average point score along the template:

$$f(T) = \frac{1}{|T|} \sum_{\mathbf{t} \in T} f(\mathbf{t}), \tag{2}$$

where $|T|$ denotes the number of sample points on part-template T. The orientation histograms are stored as a lookup table, so the computation is very similar to that of chamfer distance.

3.4 Representation Using Pose-Adaptive Descriptors

In our implementation, the global shape models (consisting of three part-template types) are represented as a set of boundary points with corresponding edge orientations. The range of the number of those model points is from 118 to 172. In order to obtain a unified (constant dimensional) description of images with those different dimensional pose models, and to establish a one-to-one correspondence between contour points of different poses (Fig. 7), we map

4. For simplicity, we use $\delta = 1$, and $w(1) = w(-1) = 0.25$, $w(0) = 0.5$ in our experiments.

the boundary points of any pose model to those of a canonical pose model. The canonical pose model is assumed to be occlusion-free so that all contour points are visible. For human upper bodies (heads and torso), the boundaries are uniformly sampled into eight left side and eight right side locations, and the point correspondence is established between poses based on vertical y coordinates and side (left or right) information. For lower bodies (legs), boundaries are uniformly sampled into locations vertically with four locations at each y value (inner leg sample points are sampled at 5 pixels apart from outer sample points in the horizontal direction). Fig. 5f shows an example of how the sampled locations are distributed).

Associated with each of those sample locations is a 36-dimensional feature vector (L_2-normalized histogram of edge orientations of its closet 2×2 block in the image). Hence, this mapping procedure generates a $(8 \times 2 + 7 \times 4) \times 36 = 1{,}584$-dimensional feature descriptor. Fig. 5 illustrates the feature extraction method. Note that only a subset of blocks is relevant to the descriptor, and a block might be duplicated several times based on the frequency of contour points lying inside the block.

4 Detecting and Segmenting Multiple Occluded Humans

Pose-adaptive descriptors discussed in the previous section are mainly developed for the purpose of detecting fully visible humans from images. However, real-world images can be crowded and it is common that humans occlude each other significantly. This is more obvious in visual surveillance scenarios, where videos are usually captured in crowded public places, e.g., shopping malls, airports, etc. In these complex cases, our generic detector based on our pose-adaptive features can be used to provide an initial set of human hypotheses (by reducing thresholds to ensure low miss rates) and then more detailed occlusion analysis and optimization can be performed.

4.1 Initial Hypotheses

The generic human detector trained using our pose-adaptive features and a discriminative classifier such as SVM can provide a reliable set of initial human hypotheses for detecting humans from still images. However, for crowded videos, a set of simpler part detectors can be more accurate than a single full-body detector due to severe occlusion between humans. Any portion of a human body can be occluded and sometimes only a very small part is visible. Hence, here, we introduce an alternative method for generating initial human hypotheses for surveillance scenarios.

Hierarchical part-template matching provides estimates of the pose model parameters for every detection window. We define a score function $F_{\mathbf{w}}$ (a function of weight vector $\mathbf{w}$) capable of evaluating image responses for any part or part combinations. The function is modeled as a weighted sum of individual part responses (matching scores) $f_j, j \in \{ht, ul, ll\}$ (computed based on (2)):

$$F_{\mathbf{w}} = \sum_j w_j f_j, \tag{3}$$

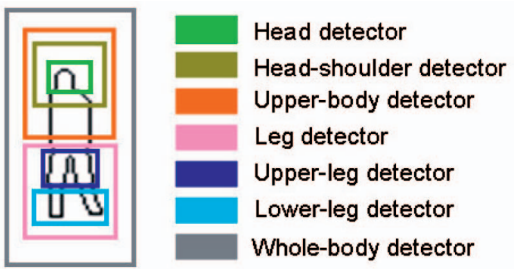


Fig. 8. An illustration of part or part combination detectors. The rectangular regions associated with different detectors are drawn with different colors. We also experimented by adding additional elementary part detectors such as head detectors and head-shoulder detectors (their detectors are built by decomposing the head-torso models vertically) since they are useful when humans are significantly occluded. Inclusion of these models improves results slightly but our experimental results are based on the simple model consisting of elementary part detectors for head-torso, upper legs, and lower legs.

where the weight distribution (or weight vector) $\mathbf{w} = \{w_j, j = ht, ul, ll\}$ $(\sum_j w_j = 1)$ characterizes different parts or part combinations. For example, $\{w_{ht} = w_{ul} = w_{ll} = 1/3\}$ corresponds to a full-body detector and $\{w_{ht} = 0, w_{ul} = w_{ll} = 1/2\}$ corresponds to a leg detector. By applying a detection threshold τ to the score function $F_{\mathbf{w}}$, we form seven part or part-combination detectors (as shown in Fig. 8), and if the head-torso is decomposed further into head-shoulder and torso, the number of detectors can be as high as 15. Suppose that we use M part detectors, D_m, $m = 1, 2 \ldots M$, corresponding to M weight vectors $\mathbf{w}_m, m = 1, 2 \ldots M$. Individual part matching score f_i is computed based on (2).

In practice, we build a pyramid from the input image and use our sliding-window-based generic human detector to reduce the search space into a small subset and boost it by searching additional hypotheses using the above part/part-combination detectors. We threshold each of the response maps (of full-body matching scores) using a constant global detection threshold τ (which is adjustable for trading off precision and recall of detections), merge nearby weak responses to strong responses, and adaptively select modes. This step can also be performed by local maximum selection after smoothing the likelihood image. The union of the maxima forms the set of human hypotheses:

$$\mathbf{u} = \{u_1, \ldots u_N\} = \{(\mathbf{x}_1, \theta^*(\mathbf{x}_1)), \ldots (\mathbf{x}_N, \theta^*(\mathbf{x}_N))\}. \tag{4}$$

We compute full-body matching scores (assuming no occlusion) for all these hypotheses and denote them by $L(u_i)$, $i = 1, 2 \ldots N$. More specifically, $L(u_i)$ is computed as the average matching score of all u_is' part-template contour points. Note that $\mathbf{u}$ is an unordered set (no occlusion ordering information).

4.2 Objective Function

Using the set of initial detection hypotheses $\mathbf{u}$ as the image observation, we model multiple occluded human detection as the problem of maximizing an objective function Φ (similar to joint likelihood maximization in [[16], 35], [36], [37]):

$$\mathbf{c}^* = \arg\max_{\mathbf{c}} \Phi(\mathbf{u}|\mathbf{c}), \tag{5}$$

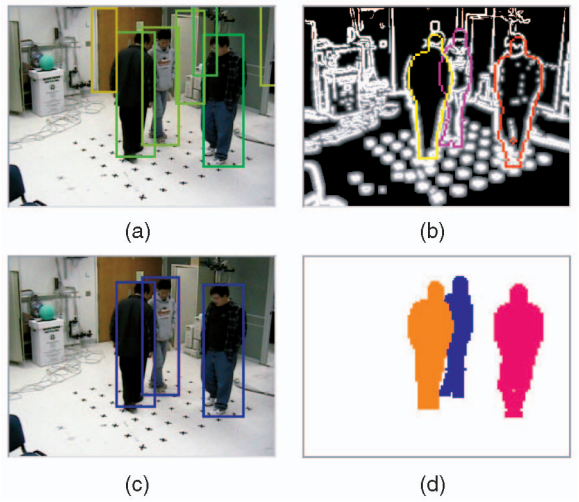


Fig. 9. An example of detection process without background subtraction. (a) Initial set of human detection hypotheses, (b) human shape segmentations, (c) detection result, and (d) segmentation result (final occlusion map).

where $\mathbf{c} = \{c_1, c_2, \ldots c_n\}$ denotes an ordered human configuration,[5] n denotes the number of humans in the configuration, and $c_i = (\mathbf{x}_i, \theta_i^*)$ denotes an individual hypothesis which consists of image location $\mathbf{x}_i$ and corresponding human model parameter θ_i^* (optimal part-template indices and locations obtained from hierarchical part-template matching). We constrain the search space to the initial set of hypotheses, i.e., $\mathbf{c} \subset \mathbf{u}$. The goal is to choose the optimal subset of initial hypotheses $\mathbf{u}$ and its occlusion ordering so that Φ is maximized.

Given an ordered human configuration $\mathbf{c}$, we can generate its occlusion map I_{occ} by overlaying regions of global shape estimates (see Fig. 9d for an example of the occlusion map). Assuming independence between each observation u_i given the configuration $\mathbf{c}$, and treating the template matching scores as pseudolikelihoods, (5) can be decomposed (similar to probability decomposition) as follows:

$$\Phi(\mathbf{u}|\mathbf{c}) = \Phi(u_1, \ldots, u_N|\mathbf{c}) = \prod_{i=1}^{N} \phi(u_i|\mathbf{c}). \quad (6)$$

Due to possible occlusions, we cannot directly use the full-body matching score $L(u_i)$ to model $\phi(u_i|\mathbf{c})$. Instead, we need to globally *reevaluate* the matching score of each hypothesis u_i based on the occlusion map I_{occ}, that is, if $u_i \in \mathbf{c}$, we compute its matching score only based on unoccluded (or visible) parts. Similarly to the occlusion handling scheme in [23], we perform occlusion compensation precisely at the pixel level via an occlusion map generated from precise human segmentations. This occlusion compensation-based score reevaluation scheme is effective in rejecting most false alarms while retaining true detections. Specifically, if $u_i \in \mathbf{c}$, there exists j such that $u_i = c_j$, and consequently, the occlusion-compensated matching score $L(u_i|I_{occ})$ is defined as the average matching score of u_i's shape template points lying inside c_j's visible region; otherwise, if $u_i \notin \mathbf{c}$, $L(u_i|I_{occ})$ is set to the constant detection threshold τ. Based on the above modeling, the objective function can be rewritten as

$$\Phi(\mathbf{u}|\mathbf{c}) = \prod_{u_i \in \mathbf{c}} L(u_i|I_{occ}) \prod_{u_i \notin \mathbf{c}} \tau. \quad (7)$$

Now, given any ordered set of human hypotheses (configuration), the objective function can be exactly evaluated.[6]

4.3 Optimization

Maximizing the objective function in (7) is an NP-hard combinatorial optimization problem. For simplifying the problem, we sort the hypotheses in decreasing order of vertical (or y) coordinate as in [16]. This is valid for many surveillance videos with ground plane assumption since the camera is typically looking obliquely down toward the scene. For notational simplicity, we assume $u_1, u_2, \ldots, u_N$ is such an ordered list. Starting from an empty set, the optimization is performed based on the iterative addition of a hypothesis to maximize the objective function in a greedy fashion. Our approach currently only adds hypotheses; although we experimented with using both addition and removal of hypotheses, it did not improve the results on our test data set. This is mainly because humans are roughly standing on a plane in these data sets. Adding hypothesis removal could be a useful extension of the algorithm for more complex occlusion cases, for example, when the ground plane assumption is not valid.

An example of the detection and segmentation process is shown in Fig. 9. Note that initial false detections are rejected in the final detection based on likelihood reevaluation, and the occlusion map is accumulated to form the final segmentation.

Algorithm 1. Optimization algorithm
Given an ordered list of initial human hypotheses $u_1, u_2, \ldots, u_N$,
initialize the configuration as $\mathbf{c}$ as the empty set, the occlusion map I_{occ} as empty (white image), and the objective function as $\Phi(\mathbf{u}|\mathbf{c}) = 0$.
for $i = 1 : N$
if $\Phi(\mathbf{u}|u_i \cup \mathbf{c}) > \Phi(\mathbf{u}|\mathbf{c})$,
(1) insert u_i to $\mathbf{c}$, i.e., $u_i \cup \mathbf{c} \mapsto \mathbf{c}$;
(2) update the occlusion map I_{occ} using the current $\mathbf{c}$;
endfor
return the configuration $\mathbf{c}$ and occlusion map I_{occ}.

5 Combining with Calibration and Background Subtraction

We can also combine the shape-based detector with background subtraction and calibration in a unified system.

5.1 Scene-to-Camera Calibration

If we assume that humans are moving on a ground plane, ground plane homography information can be estimated offline and used to efficiently control the search for humans instead of searching over all scales at all positions. A similar idea has been explored by Hoiem et al. [45] combining calibration and segmentation. To obtain a mapping between head points and foot points in the image, i.e., to estimate

5. An ordered set of hypotheses with occlusion ordering, i.e., for any $i < j$, c_j does not occlude c_i.

6. Note that L in the above equations denotes scores (also can be regarded as probabilities) but not log-probability or log-likelihood.

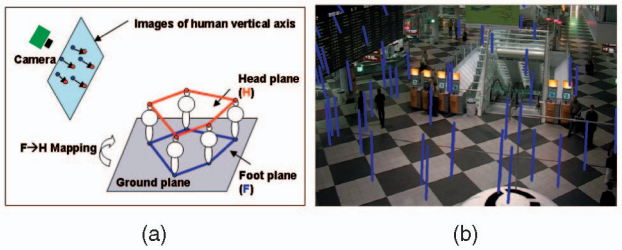


Fig. 10. Simplified scene-to-camera calibration. (a) Interpretation of the foot-to-head plane homography mapping. (b) An example of the homography mapping. Fifty sample foot points are chosen randomly and corresponding head points and human vertical axes are estimated and superimposed in the image.

expected vertical axes of humans, we simplify the calibration process by estimating the homography between the head plane and the foot plane in the image [39]. We assume that humans are standing upright on an approximate ground plane viewed by a distant camera relative to the scene scale, and that the camera is located higher than a typical person's height. We define the homography mapping as $\mathbf{f} = P_f^h : \mathcal{F} \mapsto \mathcal{H}$, where $\mathcal{F}, \mathcal{H} \in \mathbb{P}^2$. Under the above assumptions, the mapping $\mathbf{f}$ is one-to-one correspondence so that, given an offline estimated 3×3 matrix P_f^h, we can estimate the expected location of the corresponding head point $p_h = \mathbf{f}(p_f)$ given an arbitrary foot point p_f in the image. The homography matrix is estimated by the least-squares method using four or more pairs of foot and head points preannotated in some frames. An example of the homography mapping is shown in Fig. 10.

5.2 Combining with Background Subtraction

Given the calibration information and the binary foreground image from background subtraction, we estimate what we refer to as the binary foot candidate regions R_{foot} as follows: we first find all foot candidate pixels $\mathbf{x}$ with foreground coverage density $\gamma_{\mathbf{x}}$ larger than a threshold ξ. Given the estimated human vertical axis $\vec{\mathbf{v}}_{\mathbf{x}}$ at the foot candidate pixel $\mathbf{x}$, $\gamma_{\mathbf{x}}$ is defined as the proportion of foreground pixels in an adaptive rectangular window $W(\mathbf{x}, (w_0, h_0))$ determined by the foot candidate pixel $\mathbf{x}$. The foot candidate regions R_{foot} are defined as: $R_{foot} = \{\mathbf{x} | \gamma_{\mathbf{x}} \geq \xi\}$. The window coverage is efficiently calculated using integral images [2]. We detect edges in the augmented foreground regions R_{afg}, which are generated from the foot candidate regions R_{foot} by taking the union of the rectangular regions determined by each foot candidate pixel $p_f \in R_{foot}$, adaptively based on the estimated human vertical axes. Fig. 11 shows an example.

6 Experimental Results

We first present results using our generic human detector on two public pedestrian data sets and then discuss results of our multiple occluded human detector on three crowded image and video data sets.

6.1 Detection and Segmentation Using Pose-Adaptive Descriptors

We evaluate our generic human detector mainly using the INRIA person data set[7] [3] and the MIT-CBCL pedestrian

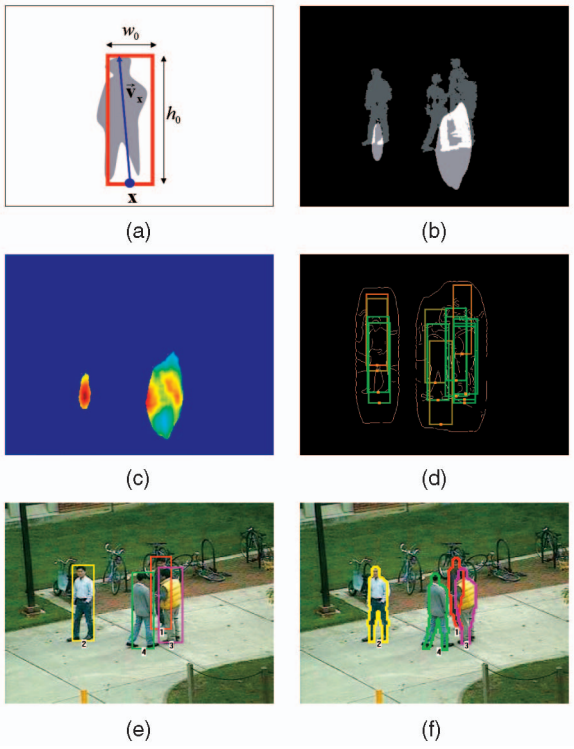


Fig. 11. An example of the detection process with background subtraction. (a) Adaptive rectangular window. (b) Foot candidate regions R_{foot} (lighter regions). (c) Object-level (foot-candidate) likelihood map by the hierarchical part-template matching (where red color represents higher probabilities and blue color represents lower probabilities). (d) The set of human hypotheses overlaid on the Canny edge map in the augmented foreground region (green boxes represent higher likelihoods and red boxes represent lower likelihoods). (e) Final human detection result. (f) Final human segmentation result.

data set[8] [1], [17]. The MIT-CBCL data set contains 924 front/back view positive images (no negative images), and the INRIA data set contains 2,416 positive training samples and 1,218 negative training images plus 1,132 positive testing samples and 453 negative testing images. Comparing to the MIT data set, the INRIA data set is much more challenging due to significant pose articulations, occlusion, clutter, viewpoint, and illumination changes.

6.1.1 Detection Performance

We evaluate our detection performance and compare it with other approaches using Detection-Error-Trade-Off (DET) curves, plots of miss rates versus false positives per window (FPPW).

Training. We first extract pose-adaptive descriptors for the set of 2,416 positive and 12,180 negative samples and batch-train a discriminative classifier for the initial training algorithm. We use the publicly available LIBSVM tool [46] for binary classification (RBF Kernel) with parameters tuned to $\mathrm{C} = 8{,}000$, $\mathrm{gamma} = 0.04$ (as the default classifier). These parameter values are estimated via twofold cross validation by evenly partitioning the original training data into validation train and validation test sets.

For improving performance, we perform one round of bootstrapping procedure for retraining the initial detector.

7. http://lear.inrialpes.fr/data.

8. http://cbcl.mit.edu/software-datasets/PedestrianData.html.

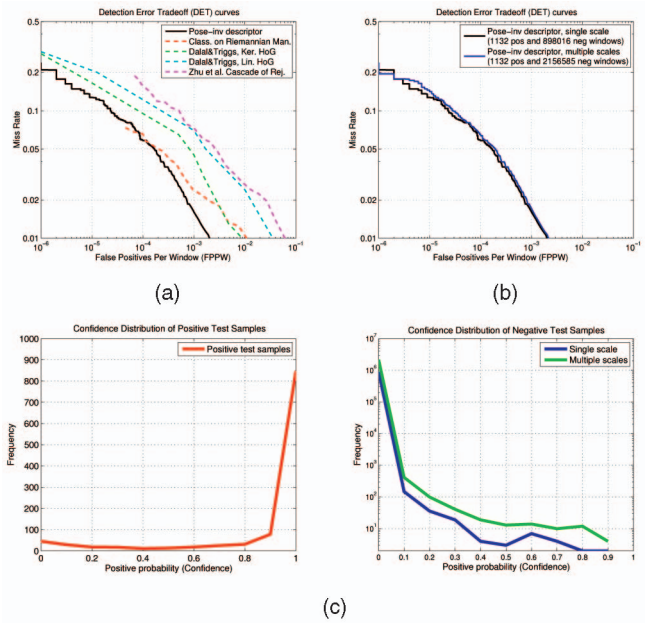


Fig. 12. Detection performance evaluation on the INRIA data set. (a) The proposed approach (testing on single scale) is compared to Kernel HOG-SVM [3], Linear HOG-SVM [3], Cascaded HOG [30], and Classification on Riemannian Manifold [5]. The results of [3] are copied from the original paper, and the results of [5], [30] are obtained by running their original detectors on the same test data. (b) Performance comparison w.r.t. the number of negative windows scanned. (c) Distribution of confidence values for positive and negative test windows.

We densely scan 1,218 (plus mirror versions) person-free photos by 8-pixel strides in horizontal/vertical directions and 1.2 scale (downsampling) factors (until the resized image does not contain any detection window) to bootstrap false positive windows. This process generates

Fig. 13. Detection results. (a) Example detections on the INRIA test images, nearby windows are merged based on distances. (b) and (c) Examples of false negatives (FNs) and false positives (FPs) generated by our detector.

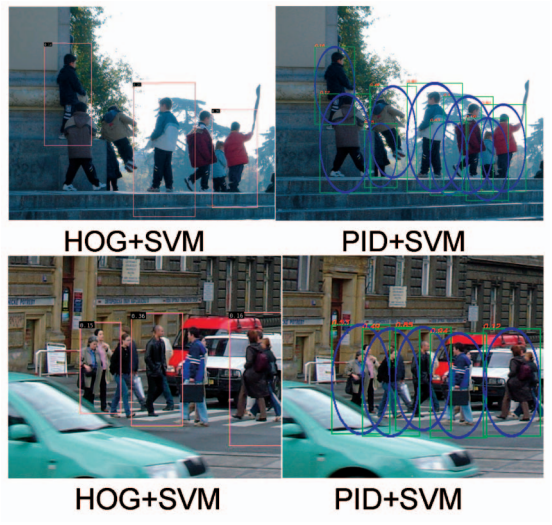


Fig. 14. Qualitative comparisons of our pose-insensitive (adaptive) descriptor (PID) with the HOG descriptor.

41,667 "hard" samples out of examined windows. These samples are normalized to 128×64 and added to the original 12,180 negative training samples and the whole training process is performed again.

Testing. For evaluation on the MIT data set, we chose its first 724 image patches as positive training samples and 12,180 training images from the INRIA data set as negative training samples. The test set contains 200 positive samples from the MIT data set and 1,200 negative samples from the INRIA data set. As a result, we achieve 1.0 percent true positive rate, and a 0.00 percent false positive rate even without retraining. Direct comparisons on the MIT data set are difficult since there are no negative samples and no separation of training and testing samples in this data set. Indirect comparisons show that our result on this data set is similar to the performance achieved previously in [3].

For the INRIA data set, we evaluated our detection performance on 1,132 positive image patches and 453 negative images. Negative test images are scanned exhaustively in the same way as in retraining. The detailed comparison of our detector with the current state-of-the-art detectors on the INRIA data set is plotted using the DET

Fig. 15. Example results of pose segmentation.

Fig. 16. Detection and segmentation results (without background subtraction) for USC pedestrian data set-B.

curves, as shown in Fig. 12. The comparison shows that our approach is comparable to the state-of-the-art human detectors. The dimensionality of our features is less than half of that used in HOG-SVM [3], but we achieve better performance. Another advantage of our approach is that it is capable of not only detecting but also segmenting human shapes and poses. In this regard, our approach can be further improved because our current pose model is very simple and can be extended to cover a much wider range of articulations. Fig. 13 shows examples of detections on whole images and examples of false negatives and false positives from our experiments. Note that FNs are mostly due to the unusual poses or illumination conditions, or significant occlusions; FPs mostly appeared in highly textured samples (such as trees) and structures resembling human shapes. Fig. 14 shows qualitative comparisons of our pose-adaptive descriptors with HOG descriptors [3] on detecting humans in natural images. Our detector successfully detected very difficult poses while the HOG-based detector missed them.[9]

6.1.2 Segmentation Performance

Fig. 15 shows some qualitative results of our pose/shape segmentation algorithm on the INRIA data set. Our pose model and probabilistic hierarchical part-template matching algorithm give very accurate segmentations for most images in the MIT-CBCL data set and on over 80 percent of 3,548 training/testing images in the INRIA data set. Significantly, poor pose estimation and segmentation are observed in about 10 percent of the images in the INRIA data set, and most of those poor segmentations were due to the very difficult poses and significant misalignment of humans.

Our detection and segmentation system is implemented in C++ and the current running time (on a machine with 2.2 GHz CPU and 3 GB memory) is as follows: Both pose segmentation and feature extraction for 800 windows take less than 0.2 second, classifying 800 windows with the RBF-Kernel SVM classifier takes less than 10 seconds, initial classifier training takes about 10 minutes, and retraining takes about two hours. The computational overhead is only due to the kernel SVM classifier, which can be replaced with a much faster boosted cascade of classifiers [2] (which we have implemented recently and which runs at three frames/second on a 320×240 image scanning 800 windows); this is comparable to [5] (reported as less than 1 second scanning 3,000 windows).

6.2 Detection and Segmentation of Multiple Occluded Humans

In order to quantitatively evaluate the performance of our detector, we use the overlap measure defined in [10]. The overlap measure is calculated as the smaller value of the area ratios of the overlap region and the ground-truth-annotated region/detection region. If the overlap measure of a detection is larger than a certain threshold $\eta = 0.5$, we regard the detection as correct.

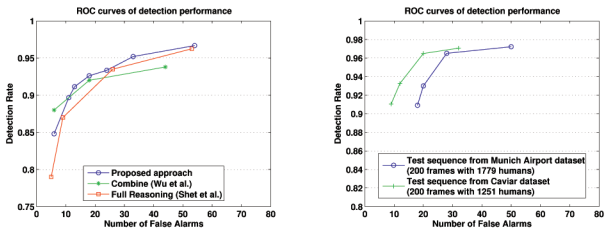


Fig. 17. Performance evaluation on three data sets. (a) Evaluation of detection performance on USC pedestrian data set-B (54 images with 271 humans). Results of [16] and [18] are copied for the comparison purpose. (b) Evaluation of detection performance on two test sequences from Munich Airport data set and Caviar data set.

9. The source codes of the generic human detector training and testing algorithms are publicly available at http://terpconnect.umd.edu/~zhelin/PidUMD.html.

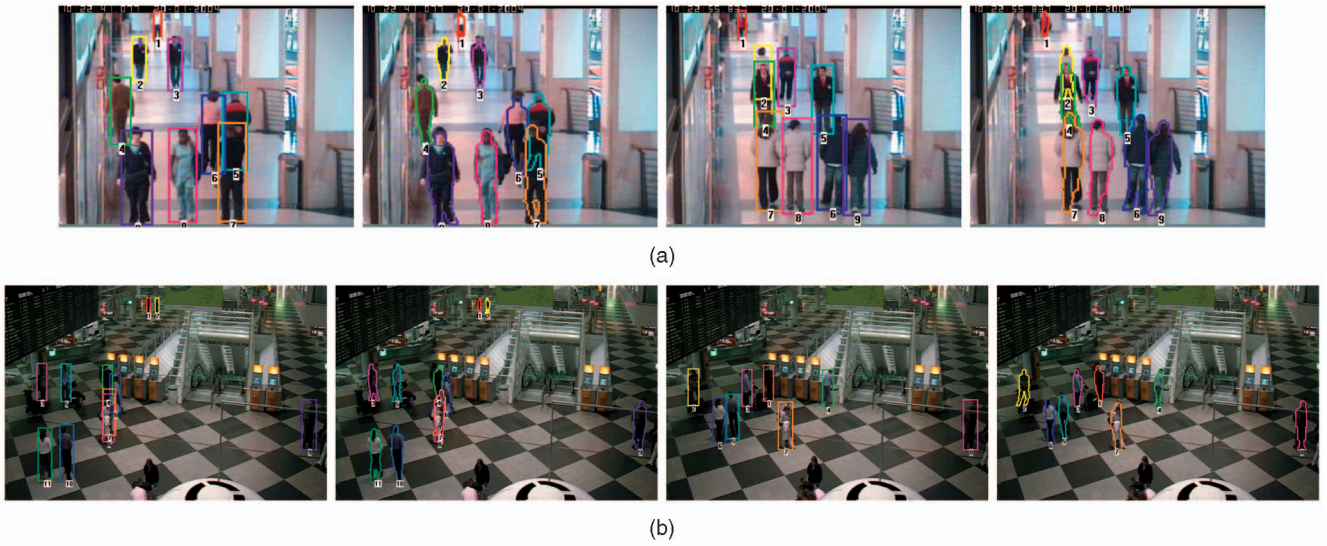


Fig. 18. Detection and segmentation results (with background subtraction) for (a) the Caviar data set and (b) the Munich Airport data set.

6.2.1 Results without Background Subtraction

We compared our human detector with Wu and Nevatia [16] and Shet et al. [18] on USC pedestrian data set-B [16], which contains 54 gray-scale images with 271 humans. In these images, humans are heavily occluded by each other and partially out of the frame in some images. Note that no background subtraction is provided for these images. Fig. 16 shows some example results of our detector and Fig. 17a shows the comparison result as ROC curves. Our detector obtained better detection performance than the others when allowing more than 10 false alarms out of total of 271 humans, while detection rate decreased significantly when the number of false alarms was reduced to 6 out of 271. Proper handling of the edge sharing problem would reduce the number of false alarms further while maintaining the detection rates. The running time of [16] for processing a 384×288 image is reported as about one frame/second on a Pentium 2.8 GHz machine, while our current running time for a same-sized image is two frames/second on a Pentium 2 GHz machine.

6.2.2 Results with Background Subtraction

We also evaluated our detector on two challenging surveillance video sequences using background subtraction. The first test sequence (1,590 frames) is selected from the Caviar Benchmark data set[10] and the second one (4,836 frames) is selected from the Munich Airport data set collected by Siemens Corporate Research.[11] The foreground regions detected from background subtraction are very noisy and inaccurate in many frames. From example results in Fig. 18, we can see that our proposed approach achieves good performance in accurately detecting humans and segmenting the boundaries even under severe occlusion and very inaccurate background subtraction. Also, from the results, we can see that the shape estimates automatically obtained from our approach are quite accurate. Some misaligned shape estimates are generated mainly due to low contrast and/or background clutter.

We evaluated the detection performance quantitatively on 200 selected frames from each video sequence. Fig. 17b shows the ROC curves for the two sequences. Most false alarms are generated by cluttered background areas incorrectly detected as foreground by background subtraction. Misdetections (true negatives) are mostly due to the lack of edge segments in the augmented foreground region or complete occlusion between humans. Our system is implemented in C++ and currently runs at about two frames/second (without background subtraction) and five frames/second (with background subtraction) for 384×288 video frames on a Pentium-M 2 GHz Machine.

6.2.3 Integration with a Region-Based Tracker

We also integrated our multiple occluded human detector with a regional affine-invariant tracker [47] and evaluated the integrated multiperson tracker on the CLEAR07 data set.[12] The detector was used to provide a reliable set of human hypotheses for visual tracking. Details of the integration method are described in [48]. Quantitative evaluation of person tracking on a large amount of challenging video data showed that our integrated detection and tracking system achieves the best result in the competition of 2D person tracking.

7 Conclusions

A hierarchical part-template matching approach is employed to match human shapes with images to detect and segment humans simultaneously. Local part-based and global shape-template-based approaches are combined to detect and segment humans from images. Based on the shape matching approach, we first introduced a pose-adaptive image descriptor for learning a discriminative classifier for the challenging problems of detecting and segmenting humans in images. The descriptor is computed adaptively based on human poses instead of concatenating features along 2D image locations as in previous approaches.

10. http://homepage.inf.ed.ac.uk/rbf/CAVIAR/.

11. The selected data can be downloaded from ftp://ftp.umiacs.umd.edu/pub/zhelin/iccv07/dataset.

12. http://www.clear-evaluation.org.

Specifically, we estimate the poses using a fast hierarchical matching algorithm based on a learned part-template tree. Given the pose estimate, the descriptor is formed by concatenating local features along the pose boundaries using a one-to-one point correspondence between detected and canonical poses. The pose-adaptive descriptors are used to train discriminative classifiers to learn generic human detectors. For applying the tree matching approach to multiple occluded human detection in crowded surveillance scenarios, we also introduced an approach to iteratively optimize the human configuration and occlusion ordering.

The results demonstrate that the proposed part-template tree model captures the articulations of the human body and detects humans robustly and efficiently. Although our approach can handle the majority of standing human poses, many of our misdetections are still due to the pose estimation failures. This suggests that the detection performance could be further improved by extending the part-template tree model to handle more difficult poses and to cope with alignment errors in positive training images. We are also investigating the addition of color and texture statistics to the local contextual descriptor to improve the detection and segmentation performance.

ACKNOWLEDGMENTS

The support of the US Defense Advanced Research Projects Agency (DARPA) under project VIRAT (subcontractor to Kitware, Inc.) is gratefully acknowledged. The authors would like to thank Siemens Corporate Research for providing the Munich Airport Surveillance Video data set for experiments.

REFERENCES

[1] C. Papageorgiou, T. Evgeniou, and T. Poggio, "A Trainable Pedestrian Detection System," *Proc. Symp. Intelligent Vehicles,* pp. 241-246, 1998.

[2] P. Viola and M. Jones, "Rapid Object Detection Using a Boosted Cascade of Simple Features," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 511-518, 2001.

[3] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 886-893, 2005.

[4] Y. Wu, T. Yu, and G. Hua, "A Statistical Field Model for Pedestrian Detection," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 1023-1030, 2005.

[5] O. Tuzel, F. Porikli, and P. Meer, "Human Detection via Classification on Riemannian Manifold," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1-8, 2007.

[6] D.M. Gavrila and V. Philomin, "Real-Time Object Detection for SMART Vehicles," *Proc. IEEE Int'l Conf. Computer Vision,* vol. 1, pp. 87-93, 1999.

[7] L. Zhao and L.S. Davis, "Closely Coupled Object Detection and Segmentation," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 454-461, 2005.

[8] D.M. Gavrila, "A Bayesian, Exemplar-Based Approach to Hierarchical Shape Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 8, pp. 1408-1421, Aug. 2007.

[9] K. Mikolajczyk, C. Schmid, and A. Zisserman, "Human Detection Based on a Probabilistic Assembly of Robust Part Detectors," *Proc. European Conf. Computer Vision,* vol. 1, pp. 69-82, 2004.

[10] B. Leibe, E. Seemann, and B. Schiele, "Pedestrian Detection in Crowded Scenes," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 878-885, 2005.

[11] E. Seemann, B. Leibe, and B. Schiele, "Multi-Aspect Detection of Articulated Objects," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 1582-1588, 2006.

[12] J. Shotton, A. Blake, and R. Cipolla, "Contour-Based Learning for Object Detection," *Proc. IEEE Int'l Conf. Computer Vision,* vol. 1, pp. 503-510, 2005.

[13] A. Opelt, A. Pinz, and A. Zisserman, "A Boundary-Fragment-Model for Object Detection," *Proc. European Conf. Computer Vision,* vol. 2, pp. 575-588, 2006.

[14] V. Ferrari, T. Tuytelaars, and L.V. Gool, "Object Detection by Contour Segment Networks," *Proc. European Conf. Computer Vision,* vol. 3, pp. 14-28, 2006.

[15] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid, "Groups of Adjacent Contour Segments for Object Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, no. 1, pp. 36-51, Jan. 2008.

[16] B. Wu and R. Nevatia, "Detection of Multiple, Partially Occluded Humans in a Single Image by Bayesian Combination of Edgelet Part Detectors," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 90-97, 2005.

[17] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-Based Object Detection in Images by Components," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 4, pp. 349-361, Apr. 2001.

[18] V.D. Shet, J. Neumann, V. Ramesh, and L.S. Davis, "Bilattice-Based Logical Reasoning for Human Detection," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1-8, 2007.

[19] B. Wu and R. Nevatia, "Simultaneous Object Detection and Segmentation by Boosting Local Shape Feature Based Classifier," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1-8, 2007.

[20] R. Fergus, P. Perona, and A. Zisserman, "Object Class Recognition by Unsupervised Scale Invariant Learning," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 264-271, 2003.

[21] H. Schneiderman and T. Kanade, "Object Detection Using Statistics of Parts," *Int'l J. Computer Vision,* vol. 56, no. 3, pp. 151-177, 2004.

[22] P.F. Felzenszwalb and D.P. Huttenlocher, "Pictorial Structures for Object Recognition," *Int'l J. Computer Vision,* vol. 61, no. 1, pp. 55-79, 2005.

[23] Y. Amit and A. Trouve, "POP: Patchwork of Parts Models for Object Recognition," *Int'l J. Computer Vision,* vol. 75, no. 2, pp. 267-282, 2007.

[24] P.F. Felzenszwalb, D. McAllester, and D. Ramanan, "A Discriminatively Trained, Multiscale, Deformable Part Model," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1-8, 2008.

[25] M.P. Kumar, P.H.S. Torr, and A. Zisserman, "Obj Cut," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 18-25, 2005.

[26] J. Winn and J. Shotton, "The Layout Consistent Random Field for Recognizing and Segmenting Partially Occluded Objects," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 37-44, 2006.

[27] P. Viola, M. Jones, and D. Snow, "Detecting Pedestrians Using Patterns of Motion and Appearance," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 734-741, 2003.

[28] N. Dalal, B. Triggs, and C. Schmid, "Human Detection Using Oriented Histograms of Flow and Appearance," *Proc. European Conf. Computer Vision,* vol. 2, pp. 428-441, 2006.

[29] V. Sharma and J.W. Davis, "Integrating Appearance and Motion Cues for Simultaneous Detection and Segmentation of Pedestrians," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 1-8, 2007.

[30] Q. Zhu, S. Avidan, M.-C. Yeh, and K.-T. Cheng, "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 1491-1498, 2006.

[31] S. Maji, A.C. Berg, and J. Malik, "Classification Using Intersection Kernel Support Vector Machines Is Efficient," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1-8, 2008.

[32] P. Sabzmeydani and G. Mori, "Detecting Pedestrians by Learning Shapelet Features," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1-8, 2007.

[33] B. Wu and R. Nevatia, "Optimizing Discrimination-Efficiency Tradeoff in Integrating Heterogeneous Local Features for Object Detection," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1-8, 2008.

[34] J. Pang, Q. Huang, and S. Jiang, "Multiple Instance Boost Using Graph Embedding Based Decision Stump for Pedestrian Detection," *Proc. European Conf. Computer Vision,* vol. 4, pp. 541-552, 2008.

[35] H. Tao, H. Sawhney, and R. Kumar, "A Sampling Algorithm for Detecting and Tracking Multiple Objects," *Proc. IEEE Int'l Conf. Computer Vision Workshop Vision Algorithms,* pp. 53-68, 1999.
[36] M. Isard and J. MacCormick, "BraMBLe: A Bayesian Multiple-Blob Tracker," *Proc. IEEE Int'l Conf. Computer Vision,* vol. 1, pp. 34-41, 2001.
[37] T. Zhao and R. Nevatia, "Tracking Multiple Humans in Crowded Environment," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 406-413, 2004.
[38] K. Smith, D.G. Perez, and J.M. Odobez, "Using Particles to Track Varying Numbers of Interacting People," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 962-969, 2005.
[39] J. Rittscher, P.H. Tu, and N. Krahnstoever, "Simultaneous Estimation of Segmentation and Shape," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 486-493, 2005.
[40] Q. Zhao, J. Kang, H. Tao, and W. Hua, "Part Based Human Tracking in a Multiple Cues Fusion Framework," *Proc. Int'l Conf. Pattern Recognition,* vol. 1, pp. 450-455, 2006.
[41] P. Dollar, B. Babenko, S. Belongie, P. Perona, and Z. Tu, "Multiple Component Learning for Object Detection," *Proc. European Conf. Computer Vision,* vol. 2, pp. 211-224, 2008.
[42] Z. Lin, L.S. Davis, D. Doermann, and D. DeMenthon, "Hierarchical Part-Template Matching for Human Detection and Segmentation," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 1-8, 2007.
[43] D. Tran and D.A. Forsyth, "Configuration Estimates Improve Pedestrian Finding," *Proc. Conf. Advances in Neural Information Processing Systems,* 2007.
[44] Z. Lin and L.S. Davis, "A Pose Invariant Descriptor for Human Detection and Segmentation," *Proc. European Conf. Computer Vision,* vol. 4, pp. 423-436, 2008.
[45] D. Hoiem, A. Efros, and M. Hebert, "Putting Objects in Perspective," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 2137-2144, 2006.
[46] C.-C. Chang and C.-J. Lin, *LIBSVM: A Library for Support Vector Machines,* http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2001.
[47] S. Tran and L.S. Davis, "Robust Object Tracking with Regional Affine Invariant Features," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 1-8, 2007.
[48] S. Tran, Z. Lin, D. Harwood, and L.S. Davis, "UMD_VDT, an Integration of Detection and Tracking Methods for Multiple Human Tracking," *Proc. CLEAR Workshop,* pp. 179-190, 2007.

Zhe Lin received the BEng degree in automatic control from the University of Science and Technology of China in 2002, the MS degree in electrical engineering from the Korea Advanced Institute of Science and Technology in 2004, and the PhD degree in electrical and computer engineering from the University of Maryland, College Park, in 2009. He has been a research intern at Microsoft Live Labs Research. He is currently working as a research scientist at the Advanced Technology Labs, Adobe Systems Incorporated, San Jose, California. His research interests include object detection and recognition, content-based image and video retrieval, and human motion tracking and activity analysis. He is a member of the IEEE and the IEEE Computer Society.

Larry S. Davis received the BA degree from Colgate University in 1970 and the MS and PhD degrees in computer science from the University of Maryland in 1974 and 1976, respectively. He is currently a professor in the Institute for Advanced Computer Studies and the Computer Science Department, as well as the chair of that department, at the University of Maryland. From 1977 to 1981, he was an assistant professor in the Department of Computer Science at the University of Texas at Austin. He returned to the University of Maryland as an associate professor in 1981. From 1985 to 1994, he was the director of the University of Maryland Institute for Advanced Computer Studies. He is known for his research in computer vision and high-performance computing. He has published more than 100 papers in journals and has supervised more than 20 PhD students. He is an associate editor of the *International Journal of Computer Vision* and an area editor for *Computer Models for Image Processing: Image Understanding*. He has served as the program or general chair for most of the field's major conferences and workshops, including the Fifth International Conference on Computer Vision, the 2004 Computer Vision and Pattern Recognition Conference, the 11th International Conference on Computer Vision. He is a fellow of the IEEE and a member of the IEEE Computer Society.