Integrating Appearance and Motion Cues for Simultaneous Detection and Segmentation of Pedestrians *

Vinay Sharma James W. Davis
Dept. of Computer Science and Engineering
Ohio State University
Columbus OH 43210 USA
{sharmav,jwdavis}@cse.ohio-state.edu

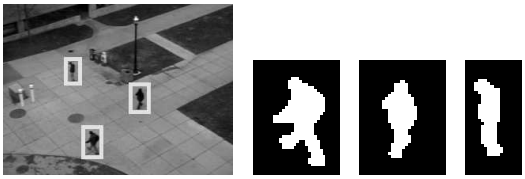
Abstract

We present a unified method for simultaneously acquiring both the location and the silhouette shape of people in outdoor scenes. The proposed algorithm integrates top-down and bottom-up processes in a balanced manner, employing both appearance and motion cues at different perceptual levels. Without requiring manually segmented training data, the algorithm employs a simple top-down procedure to capture the high-level cue of object familiarity. Motivated by regularities in the shape and motion characteristics of humans, interactions among low-level contour features are exploited to extract mid-level perceptual cues such as smooth continuation, common fate, and closure. A Markov random field formulation is presented that effectively combines the various cues from the top-down and bottom-up processes. The algorithm is extensively evaluated on static and moving pedestrian datasets for both detection and segmentation.

1. Introduction

We present a unified approach for simultaneously recovering both the *location* and the silhouette *shape* of pedestrians from outdoor scenes. The technique exploits both appearance and motion cues within a simple learning scheme utilizing cropped images of positive and negative examples. The algorithm does not require any manually marked shape information in the form of silhouettes, bounding contours, or the location of object centroids in full images.

State-of-the-art person detection algorithms [3, 11] typically provide only the location and scale of instances of the target object (bounding boxes). Though useful, this output lacks any information regarding the silhouette shape of each detected person, a cue of critical importance for higher level analysis of articulated objects (*e.g.*, action recognition). Acquiring object shape is often viewed as a distinct segmentation task, typically requiring algorithms trained on datasets consisting of manually annotated silhouette images.



(a) (b) (c) (d)

Figure 1. An example result. (a) Image showing detections using bounding boxes. (b)-(d) Extracted person silhouettes.

The proposed approach precludes the need for employing separate, sequential processes to obtain person silhouettes from prior detections. Further, our method provides shape information using training data similar to that used by detection-only algorithms, and thus does not require any manual effort towards explicitly describing object shape.

While detection and segmentation techniques typically adopt either a top-down or a bottom-up methodology, our technique strives towards integrating both processes in a balanced manner, exploiting appearance and motion information at different perceptual levels. At the lowest level, we extract edge-based contour tokens that are encoded using compact feature descriptors. The top-down module provides the high-level cue of object familiarity, utilizing the training dataset to generates likelihoods over the feature space for both the positive (person) and negative (non-person) classes. For a given input image, the bottom-up module leverages local interactions among the extracted features in order to obtain mid-level perceptual cues such as continuity, closure, and common-fate.

In order to integrate the top-down and bottom-up information, we employ a Markov random field (MRF) defined on the extracted features. The MRF is designed to optimally

*Appears in *IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, October, 2007

(in a Bayesian sense) combine the shape familiarity cues learned from the training dataset with a prior that enforces the simple observation that people have shapes that are regular and closed, and have motion that is locally coherent. Solving the MRF provides a binary labeling of the contour features, assigning each feature to either the target object (person) or the background. Local and global properties of the contour features labeled as belonging to the target are then utilized to make a decision about the presence of the object, and ultimately to form the object silhouette.

We provide in-depth experimental evaluation of the proposed method for the *detection* and *segmentation* of both *static* and *moving* people utilizing two challenging datasets (INRIA , OSU). We are able to generate high quality silhouettes on the INRIA dataset while simultaneously maintaining detection rates comparable to state-of-the-art detection-only algorithms. Using the OSU moving pedestrian dataset, we demonstrate the ability of our algorithm to combine motion information along with appearance cues to detect and segment moving and static people. Using a systematic experimental procedure, the algorithm is shown to be fairly insensitive to the relative number of moving and static examples in the training dataset.

2. Related Work

Several methods utilizing appearance and motion cues have been shown to be effective at detecting people in outdoor scenes [11, 3]. These top-down approaches, however, do not provide any information regarding object shape.

In [10] an image parsing framework was presented that provided the location and the coarse shape of target objects. In [5] a template matching approach was described that provided coarse shape information of detected pedestrians. Template-base methods, however, require large training sets with completely segmented object regions. Several attempts at recovering object shape and location have been based on implicit shape models [7]. This class of techniques utilizes spatial distributions of prototypical image patches around the object centroid and typically requires fully segmented object regions during training. In a related approach [8], discriminative boundary fragments (instead of image patches) were used to learn the object geometry. Another approach using object boundaries was proposed in [9]. In [6], an approach using a MRF defined over different object parts was proposed for detection and segmentation. All of these methods employ complex training schemes and require some form of manual segmentation during training (object centroids in [8], complete shape in [7, 9, 6]).

We propose a unified approach for simultaneous detection and segmentation utilizing only "weakly labeled" training data (Sect. 4). Unlike other detection approaches that provide object shape [9, 8], our algorithm is capable of exploiting both appearance *and* motion cues. We also extensively evaluate our method in terms of both detection and silhouette segmentation utilizing a standard static person dataset and another dataset consisting of moving people.

3. Contour Features

First-order gradient information has been often used for both object detection and segmentation. We exploit gradient information by extracting nearly linear contour fragments, formed of connected pixels having similar edge orientation. In order to ensure that the extracted contours are of reasonable size, the edge orientations are quantized into a small number of bins.

Each contour fragment is denoted by a feature vector, f. The attributes of the feature vector are chosen such that the set of all features $F = \{f_1, \ldots f_n\}$ extracted from an image effectively captures the underlying image structure. While several different attributes could be included in the descriptor, in this work we require a feature that captures local properties of image gradients such as location, extent, orientation, and magnitude. Additionally, if available, we also capture local motion information along the image gradients. We thus define the feature vector as $f = [p_1, p_2, E_{mag}, v_x, v_y]$, where the contour end-points p_1 and p_2 capture the location, extent, and orientation of local gradient information, E_{mag} denotes the mean gradient magnitude along the contour, and v_x, v_y denote the mean x- and y-components of image flow along the contour. The contour features are computed from the results of Canny-edge detection, and optical flow vectors are obtained from image pairs using the technique proposed in [1]. In Fig. 2(b) we show the contour features extracted from an example image shown in Fig. 2(a).

Given all the contour features extracted from an image region, we aim to identify those that belong to the target object (if any). An attractive property of these contour-based features is that they enable the seamless application of both top-down and bottom-up processing.

4. Top-Down Processing

The top-down process of our algorithm attempts to acquire a rough estimate of the target object shape from weakly labeled training data. The training data required is similar to that typically used by object detection algorithms, consisting of cropped images divided into two sets, a positive set containing instances of the target object, and a larger negative set not containing the object. No manual annotation in the form of segmented foreground (object) pixels is required. The dataset is weakly labeled in the sense that, for each training image, the presence or absence of the target object is labeled, but the shape of the object in the image is not marked.

We first extract features, f, as described in Sect. 3 from

each cropped image in the training set. These features populate a 7D space, where the dimensions represent the x and y coordinates of p_1 and p_2, E_{mag}, v_x, and v_y. In this 7D space, we create probability density functions (pdfs) for the positive and negative features using normalized histograms (other non-parametric techniques, such as kernel density estimation, could also be employed).

The modes of the positive pdf correspond to contour features characteristic of the target object class as seen in the training set. Given a new feature, the positive and negative pdfs are used to provide a likelihood measure of the feature belonging to the target object or the background. We note that the described training approach treats contour features independent of each other. While this potentially dilutes the benefits that a purely top-down (detection-only) approach might glean from a dataset, treating the contour features in this manner enables the proposed approach to incorporate bottom-up cues with top-down processing. Furthermore, as we will show (Sections 8.1.1 and 8.2), this independent treatment enables the approach to degrade gracefully under severe occlusion, and also to effectively extract shape cues from training data consisting of only moving pedestrians.

Based on the learned probability density functions, we identify a set F' of *candidate* contours that are more likely to belong to the object class (l_o) than the background (l_b) using the thresholded log-likelihood ratio

$$f_i \in F', \text{ if } \ln\left(\frac{p(f_i|l_o)}{p(f_i|l_b)}\right) > T \tag{1}$$

Due to the large variability in object pose, and the use of only weakly labeled data, the contour features F' determined to belong to the object at this stage are not always accurate or complete. The selected contours can potentially miss large portions of the object boundary, and also correspond to edges belonging to background structure. For example, in Fig. 2(c) we show the candidate contour features selected as belonging to the person for the image shown in Fig. 2(a). However, when this top-down information is combined with a strong bottom-up component, the algorithm is able to identify additional contours along the object boundary and discard background contours.

5. Bottom-Up Processing

In the bottom-up component of our algorithm, we attempt to capture the expectation that people (or the target object) have a natural structure with shapes that are regular and bounded, and move in a manner that is at least locally coherent. We examine the interactions between contour features to exploit both local and global perceptual cues, such as smooth continuity, common fate, and closure.

5.1. Local Interaction: Contour Affinity

We define a local interaction term based on the notion of "affinity", originally used in computational figure completion methods. Since objects tend to have locally smooth, regular boundaries, we require that the affinity between contour features lying along the same edge structure be stronger than others. We thus define contour affinity such that the features have a higher affinity if they are in close proximity, have similar orientation, and have similar edge intensity

$$\mathit{Aff}(f_1, f_2) = e^{(-r/\sigma_r)} \cdot e^{(-\beta/\sigma_t)} \cdot e^{(-\Delta/\sigma_e)} \tag{2}$$

where r is distance between end-points of contour features f_1 and f_2, and $\Delta = |E_{mag}^{f_1} - E_{mag}^{f_2}|$. The term $\beta = \theta_1^2 + \theta_2^2 - \theta_1 \cdot \theta_2$, where θ_1 denotes the angle between the tangent vector at the end-point of f_1 and the line joining the end-points of f_1 and f_2; the angle θ_2, formed at the end-point of f_2, is analogous to θ_1. The normalization factors σ_r, σ_t, and σ_e are written as $\sigma_r = R/w_1$, $\sigma_t = T/w_2$, and $\sigma_e = E/w_3$, where R, T, and E equal the maximum possible value of r, β, and Δ, and (w_1, w_2, w_3) are weights that can be used to change the relative influence of each term in the affinity calculation. Since f_1 and f_2 have two end-points each, there are four curves connecting the contours depending on which pair of end-points is connected. We define the contour affinity, $\mathit{Aff}(f_1, f_2)$, between contours f_1 and f_2 as the maximum affinity over the four possible curves.

Motion information, if available, provides another perceptually salient local cue. Based on the notion of common fate, contour features that exhibit similar motion characteristics are likely to lie along the same object boundary. In the presence of motion information, we thus update our definition of contour affinity as follows

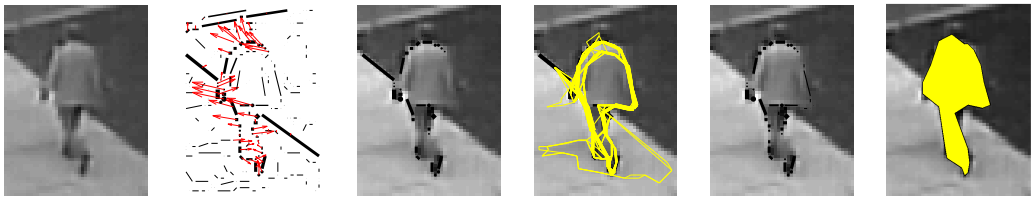
$$\mathit{Aff}(f_1, f_2) = \mathit{Aff}(f_1, f_2) \cdot e^{(-D \cdot \phi/\sigma_m)} \tag{3}$$

where ϕ is the sine of the angle between the motion direction of f_1 and f_2 and D is the difference in motion magnitude. The normalization factor $\sigma_m = 1/w_4$, where w_4 is a parameter that can be adjusted to define the contribution of the motion term to the rest of the affinity calculation.

5.2. Global Interaction: Grouping

Using the locally computed affinity values as a measure of similarity, we attempt to group the extracted contour features in a meaningful manner. Observing that objects of interest have a finite extent bounded by a closed boundary, we rely on the perceptually salient global cue of closure to guide the grouping process.

In order to compute closure, similar to [4], we treat the contour features f_i as nodes in a weighted, directed graph, where the weights on the arcs correspond to the affinity between the nodes. We limit the out-degree of each node so as



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

Figure 2. Different processing steps. (a) Input image. (b) Contour features extracted from image. (b) Candidate features (without flow vectors). (c) Cycles connecting pairs of candidate features. (d) Final selected contours. (e) Silhouette formed from selected contours.

to create a sparse graph, $\mathcal{G}$. Considering each node in turn, we compute the mean and standard-deviation of its affinity values with every other node. We then preserve only those arcs that have affinity values with a Mahalanobis distance greater than a threshold. We use a threshold value of 1 standard deviation for all the results reported here. The arcs of the graph are then assigned weights equal to the negative log of the affinity values (high affinity corresponds to low arc weight). This enables us to find the most likely cycle passing through a pair of contours using standard and efficient shortest-path algorithms (e.g., Dijkstra's algorithm). We assign to each computed cycle a score, S, equal to the product of the area of the cycle and the affinity of the arc with the maximum weight (minimum affinity) in the cycle. Thus, large cycles formed by chains of high affinity contour features are assigned higher scores.

We only search for cycles connecting pairs of contour features taken from F' (Eqn. 1). As we find cycles, C_{ij}, connecting feature f_i with other features f_j in F', we increment a pairwise interaction term, $Cyc(f_i, f_k)$, for all contours c_k included in those cycles

$$Cyc(f_i, f_k) = Cyc(f_i, f_k) + \begin{cases} S(C_{ij}) & f_k \in C_{ij} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

The value of $Cyc(f_i, f_k)$ is normalized by the number of contours in F'. Thus, a high value of $Cyc(f_i, f_k)$ suggests that, among cycles computed between contours in F', many high scoring cycles passed through f_i and f_k. For example, we show all cycles computed between the candidate contours (Fig. 2(c)) overlaid on the input image in Fig. 2(d).

6. Cue Integration

Given a rectangular image region, we begin by extracting contour features $F = \{f_1, f_2, \ldots f_n\}$ from the input image, and aim to obtain a segmentation by assigning each feature a label from the set $\mathcal{L} = \{l_o, l_b\}$, corresponding to the "object" or "background" class.

Let B denote a configuration of labels such that $\{f_1 = b_1, f_2 = b_2, \ldots f_n = b_n\}$, where $b_i \in \mathcal{L}$. We formulate the search for the optimal label configuration B as a maximum a posteriori (MAP) problem. If we assume that the likelihood of a configuration of labels can be written as a product of the individual likelihoods, the MAP estimate is equivalent to minimizing the free energy [2]

$$E(B) = -\sum_{i}^{n} \log(p(f_i|b_i)) - \log(p(B)) \quad (5)$$

The first term corresponds to the likelihood of each contour feature belonging to the positive (object) or the negative (background) class. These likelihoods are learned during the top-down training procedure, and capture the coarse shape of the specific object category. The second term corresponds to the prior probability of a shape, as defined by a given configuration of contour labels. We enforce this prior utilizing the mid-level cues obtained from the bottom-up component.

6.1. Structure and Neighborhood

We model the prior by employing a novel MRF defined over the set of contour features. In order to establish a neighborhood system for the MRF, we make use of contour affinity (see Eqn. 2) as a distance measure.

The proposed MRF is defined over an irregular, non-uniform set of elements, hence it is not feasible to employ methods that assume the presence of a regular lattice (4- or 8-connected neighborhoods). We require the neighborhood of a contour feature to be determined based on its local interactions with other features in the field. Since the structure of the graph $\mathcal{G}$ (defined in Sect. 5.2) is determined based on the distribution of affinity values, we define the neighborhood $\mathcal{N}_p$ of a contour feature f_p to be identical to the neighborhood of f_p in the graph $\mathcal{G}$ (Sect. 5.2).

6.2. Clique Potential

Following the Hammersley-Clifford theorem, we define the probability of a configuration $P(B) \propto \exp(-\sum_k V_k(B))$, where V_k denotes the clique potential defined over cliques k. We employ the generalized Potts model to define pairwise clique potentials as

$$V_{(p,q)}(b_p, b_q) = u_{(p,q)}(1 - \delta(b_p - b_q)) \quad (6)$$

Figure 3. Examples of segmented silhouettes from INRIA dataset.

where p and q are neighboring sites in the field, which, in our case denote contour features.

The MRF described here is non-homogeneous, in that the clique potential across neighboring sites (contour features) depends on the properties of the sites. Instead of defining radially symmetric clique potentials, we wish to enforce directional smoothness to the label configuration, such that if a contour feature has a positive (object) label, neighboring contours are assigned the same label only if they exhibit good continuity (high affinity) and closure (belong to a closed chain of contours).

We thus combine both *Aff* and *Cyc* in order to define the penalty term $u_{(p,q)}$ in Eqn. 6 as

$$u_{(p,q)} = \begin{cases} \mathit{Aff}(p,q) \cdot e^{(-\sigma_c / \mathit{Cyc}(p,q))} & f_p \in F' \\ \mathit{Aff}(p,q) & \text{otherwise} \end{cases} \quad (7)$$

where σ_c is a normalization constant. Thus, the cost of label discontinuity is greater for contour pairs with high affinity values. Furthermore, if a contour is in F', this cost is greater for those pairs that have a high affinity and are likely to belong to a closed contour cycle.

6.3. Energy Minimization

As shown in [2], minimizing the energy function $E(B)$ in Eqn. 5 is equivalent to solving the mincut problem on an appropriately constructed graph. Following [2], the graph is composed of two types of vertices, the c-vertices (contour features) and the l-vertices (labels, l_o and l_b). Among the c-vertices, if q is in the neighborhood of p, then p and q are connected by an arc with weight $w_{(p,q)} = 2u_{(p,q)}$. Each c-vertex also has an incoming directed arc from l_o (source) and an outgoing directed arc to l_b (sink) with a weight

$$w_p^l = (\ln(P(f_p|l)) + K) + \sum_{q \in \mathcal{N}_p} w_{(p,q)} \quad (8)$$

where $l \in \mathcal{L}$ and K is a constant ensuring that the weights are positive. The min-cut of this graph ensures that each contour feature is connected to only one of the l-vertices, l_o or l_b, and provides the required contour labeling. In Fig. 2(e) we show the final person contours obtained using the proposed MRF for the image shown in Fig. 2(a).

7. Detection and Silhouette Segmentation

The labels assigned to the contours in an image region directly enable us to classify the input region as containing the target object or not. Typically, if the target object is not present, all the contour features are assigned label l_b (background) and the classification of the image region follows trivially. However, depending on the structure in the scene, it is possible that some background contour features are incorrectly assigned label l_o. Hence, in order to correctly classify an image patch we employ a simple measure of the "coherence" of the contours labeled l_o.

We first search for cycles formed by the positively labeled contours, scoring them based on the area enclosed by the cycle and the minimum affinity arc in the cycle (as described in Sect. 5.2). The coherence measure is defined as the average of the score of the best cycle and the median positive likelihood of the contours in the cycle. The coherence measure ensures that a segmented region is assigned a high score only if both the top-down and bottom-up components provide ample support to its bounding contours.

A simple threshold applied to the coherence measure serves as the final classifier to determine if the detection window contains the target object or not. Alternately, based on domain knowledge and prior expectation of the target object shape, more complex shape-based classification schemes could also be applied at this stage. Traditional object detection schemes, that merely classify image patches without providing a segmentation of the object region, do not provide any such opportunity to reason about falsely classified image patches.

We note that computing coherence does not impose a large computational overhead since the pairwise contour affinities are already computed, and the graph is sparse, consisting only of the positively labeled contours. Computing the coherence measure also directly provides us with the silhouette of the object, obtained by simply connecting the end-points of the contours along the highest scoring cycle, and then flood-filling the resulting closed outline. In Fig. 2(f) we show the final silhouette obtained using this method from the selected person contours (shown in Fig. 2(e)).

8. Experiments

We evaluate our algorithm on two challenging datasets and provide an in-depth analysis of the detection and segmentation performance for both stationary and moving people. First, we make use of the well established INRIA person dataset consisting of static images of people in various upright poses. While this dataset was originally intended for evaluating person detection algorithms, we aim to utilize the same dataset to not only detect people, but also to acquire a segmentation of the person shape.

Second, we employ a dataset consisting of images of

moving people taken from several different surveillance cameras on the Ohio State University (OSU) campus. The OSU dataset is organized similar to the INRIA dataset, and does not contain any explicit information regarding person shape (no marked silhouettes or contours are provided for training). In what follows we provide a detailed analysis of the performance of our algorithm for the detection and segmentation of people using these two datasets.

8.1. Static People

To train our system on the static INRIA person dataset we ignore the flow components of the feature vectors extracted from the images. The positive contour pdfs were obtained from the 2478 positive training examples cropped to a size of 60×120. For the negative contour pdf, the algorithm was first trained on 12180 image patches chosen at random from the 1218 negative training examples. Similar to [3], additional harder examples were obtained using a single stage of boot-strapping. We employed a Canny edge detector with $\sigma = 1.2$ and an orientation bin size of 45° to extract the contour features.

The test set consists of 1126 positive examples and 453 full negative images not containing people. As specified in [3], the negative set was scanned for false-positives at multiple resolutions using a 4-level image pyramid. For the *evaluation* of silhouette segmentations provided by our algorithm, we also augmented this dataset with hand-drawn silhouettes for 600 person examples chosen at random from the positive test set.

We experimented with several different combinations of bin numbers for the spatial and edge magnitude dimensions. Best results were obtained for settings with a low spatial resolution and a high resolution for the edge magnitude, namely 15 and 30 bins in the x and y direction respectively, and 8 bins for edge magnitude.

Similar to [3], we evaluated the detection performance of our algorithm by comparing the miss-rates at different FPPW (false positives per window) values. To evaluate the segmentation results at these levels of detection, we compared the generated silhouettes against the ground-truth, and computed the F-measure of Precision and Recall to determine the level of overlap at the pixel-level.

In Fig. 3 we show examples of segmented silhouettes obtained from the INRIA dataset. Table 1 summarizes both the detection and segmentation results achieved by our algorithm at three different FPPW values. The F-measure for the segmentation results are computed only for the correctly detected examples. The results presented in the table show that the proposed algorithm is capable of high detection rates while ensuring only a small number of false positives. As a comparison, we also present in the table the detection results achieved by the HOG detector [3]. At an FPPW of 1×10^{-4}, we see that our algorithm has a slightly higher miss-rate than the HOG detector. However, the purely top-down approach of the HOG algorithm is not capable of providing any information regarding the silhouette shape of the detected people. In contrast, the combined top-down and bottom-up approach of the proposed algorithm provides fairly good quality silhouettes, while maintaining reasonable detection rates. At higher FPPW values of 2×10^{-4} and 2.5×10^{-4} we see from Table 1 that our algorithm outperforms the HOG detector even in terms of detection performance while continuing to maintain the quality of silhouette segmentation.

	Proposed		HOG	
FPPW	Miss rate	F-msr.	Miss rate	F-msr.
1×10^{-4}	0.23	0.82	0.11	-
2×10^{-4}	0.08	0.80	0.09	-
2.5×10^{-4}	0.00	0.80	0.09	-

Table 1. Miss rate (detection) and F-measure (segmentation) at different FPPW values for the proposed algorithm and HOG.

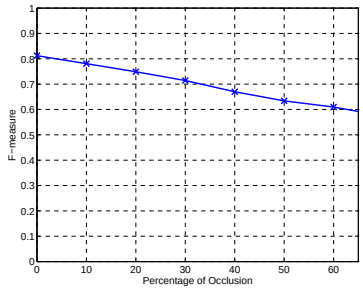


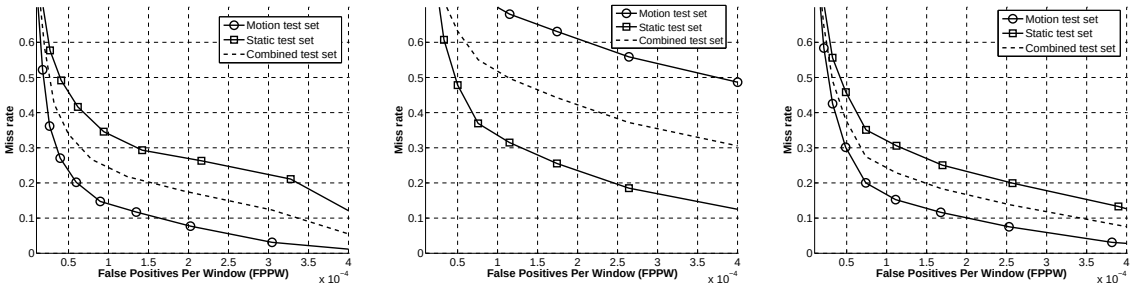
Figure 4. Effect of occlusion on silhouette quality (INRIA dataset).

8.1.1 Occlusions

We also studied the effect of occlusions on the ability of the algorithm to extract person silhouettes. We first randomly selected several hundred image patches of different sizes from various images not containing people. These image patches were superimposed at random locations on each of the 600 positive test images of the INRIA dataset for which ground truth silhouettes were available. Based on the ground truth information, the percentage of occlusion for each image was systematically increased from 0% in increments of 10%. The plot in Fig. 4 shows the F-measure of segmentation performance for each occlusion level averaged over 10 runs of the experiment. As can be seen from the plot, the performance shows only a gradual decline in performance for even up to as much as 50% occlusion.

8.2. Static and Moving People

We next present a systematic evaluation of how our algorithm combines motion cues with appearance for the tasks of detecting and segmenting moving and static peo-



(a) (b) (c)

Figure 5. Moving and static person detection using different training sets. (a) Motion only. (b) Static only. (c) Motion and static combined.

ple. We make use of the OSU moving person dataset containing pairs of images taken from successive video frames (recorded at 15Hz). This dataset poses several challenging conditions not seen in the previous INRIA dataset. The images correspond to typical outdoor surveillance scenarios comprised of low-resolution (320×240) images captured from viewpoints at several different elevations, with no constraint on the pose of the person in the image.

The positive and negative training sets consist of 45×60 image pairs with 2388 and 11434 examples, respectively. The positive test set consists of 1200 image pairs of the same size (including left-right reflections). The negative test-set is comprised of 180 pairs of 320×240 images not containing people, and is exhaustively scanned using a 2-level image pyramid (scale-factor of 0.8) for false positives. In order to be able to assess segmentation performance, we also manually marked silhouettes for 830 examples chosen at random from the positive test set.

All the positive (training and testing) examples in this dataset correspond to pedestrians exhibiting natural motion in the scene. Also, care was taken to ensure that the negative examples consist of both static and moving non-person examples. Using the sum of the motion magnitude in an image patch as a simple measure, we found that 32% of the negative training examples had at least as much motion as the positive example with the smallest motion. Examples of non-person motion captured in our dataset include vehicles, cyclists, smoke, trees and shrubs, and camera jitter.

The ultimate goal is to be able to detect and segment people irrespective of whether they are moving or stationary. The proficiency of an algorithm for dealing with static or moving people is likely influenced by the composition of such examples in the training data. Ideally, however, one would hope that an algorithm is only minimally sensitive to the nature of the training data, and that it is able to effectively capture both target appearance and motion cues without requiring a carefully balanced dataset.

In order to study the sensitivity of our algorithm to the composition of the training data, we trained our algorithm on three different subsets of positive images:

Motion-only training set (M): consisting of all 2388 image pairs in the positive training set.

Static-only training set (S): consisting of only the first frame of each of the 2388 examples in the positive training set (no motion information).

Motion and static combined training set (M+S): consisting of one half (chosen randomly) of the examples from the positive training set with motion information, and the other half without motion.

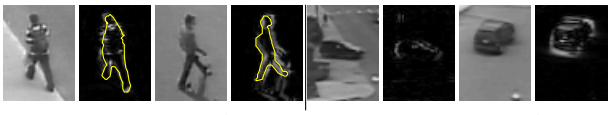
Splitting the training sets as described above ensures that the size of the training set remains the same and eliminates any potential bias in performance resulting from access to more training data. The algorithm trained on each of the above image sets was evaluated on moving and static (by considering only the first frames of each example) examples in the test set.

Figure 5 shows the Detector Error Tradeoff (DET) plots fore each version of our algorithm (S, M, and S+M) for the detection of static and moving people. The plots also show (using a dashed line) the detection performance averaged over moving and static people. Several interesting aspects emerge from examining these plots.

First, comparing the average curves from Fig. 5(a) and Fig. 5(b), we note the clear improvement in performance obtained by including motion information. At 1×10^{-4} FPPW, inclusion of motion information corresponds to a reduction in miss-rate by 27.5% as compared to a purely static detector. From Fig. 5(a) we see that version M, trained exclusively on moving pedestrians, is able to generate good results for detecting both moving and static people. Comparing the individual curves in Figs. 5(a) and (b), we see that the performance of version M for the detection of stationary people is very comparable to that of version S, trained exclusively on static examples. Further, we also note the similarity in the average curves from Figs. 5(a) and (c). The above results suggest that the proposed algorithm is indeed fairly robust to the composition of the training set, and is able to effectively capture appearance cues of the target object even from a dataset consisting of only moving objects. This also explains why the presence of static ex-

	Motion	Static	Average
M	78.85	77.67	78.26
S	68.60	78.10	73.35
M+S	79.02	78.40	78.71

Table 2. Segmentation F-measure values at 1×10^{-4} FPPW.



(a) (b) (c) (d)

Figure 6. Examples of correct classification from the OSU dataset. (a)-(b) Person. (c)-(d) Non-person. (see text for details)

amples in the M+S dataset does not contribute significantly to detection performance. As we will discuss in Sect. 8.3, this useful capability of our algorithm is afforded by our independent treatment of contour features.

The segmentation performance of the three versions of our algorithm follow the same trend shown by the detection results discussed above. The silhouette quality for the correctly detected people at a FPPW of 1×10^{-4} is summarized in Table 2. We see that while version M+S provides the best quality silhouettes, the segmentation results of version M are indeed very similar to that of the combined M+S version for both static and moving people.

Figure 6 shows sample results from the OSU dataset corresponding to person and non-person image regions. For each example, we show one image from the input image pair and also the result of image differencing (motion) across the input image pair. The detected object silhouette, if any, is overlaid on the difference image.

8.3. Discussion

The above experiments on static and moving detection and segmentation bring forth the benefits of closely incorporating a top-down technique with a strong bottom-up component. In the comparison with the HOG detector (Sect. 8.1, Table 1), while the pure top-down strategy of the HOG detector generated a better detection rate at 1×10^{-4} FPPW, it is unable to provide any shape infomation. Also, it is worth noting that at slightly higher false-positive rates, we achieve much lower miss-rates than the HOG detector. This is because exploiting bottom-up cues enables our method to correctly segment, and hence discount, the majority of windows scanned in an image. Apart from being able to acquire object shape, such an approach provides other advantages over purely top-down methods, as described below.

A balanced approach to top-down and bottom-up processing enables a graceful degradation of performance over different levels of occlusion (Sect. 8.1, Fig. 4). A purely top-down approach will be unable to handle such a wide variety of occlusions unless specifically trained to do so. Treating the contour features independently enables the top-down process in our algorithm to provide valuable cues even when only parts of the body are visible. This, together with bottom-up processing, enables our method to obtain a reasonable result in the face of occlusion.

A crucial benefit of treating features independently during top-down processing, followed by a strong bottom-up component, is evident from the results obtained by our algorithm trained purely on moving pedestrians (Sect. 8.2, Fig. 5(a), Table 2). While the training data contains no examples of stationary people, the algorithm is able to "piece together" appearance cues from the different parts of the body that remain almost static during natural human motion. Effectively combining strong perceptual cues, our algorithm is able to utilize this information to both detect and segment stationary people.

9. Conclusion

We presented a method for the simultaneous detection and segmentation of people using appearance and motion cues using only weakly labeled data. The method integrates both top-down and bottom-up processing in a coherent manner. An MRF formulation defined over contour features is employed to integrate these different sources of information and provide a contour labeling. We extensively evaluated our method using two challenging datasets. The results demonstrate the ability of the approach to effectively detect and segment both moving and stationary people. The algorithm is reasonably robust to occlusion, and has the desirable property of being insensitive to the relative composition of moving and stationary people in the training dataset.

10. Acknowledgments

This research was supported in part by the National Science Foundation under grant No. 0428249.

References

[1] M. Black et al. A framework for the robust estimation of optical flow. In *Proc. ICCV*, pages 231–236, 1993.

[2] Y. Boykov et al. Fast approximate energy minimization via graph cuts. *IEEE TPAMI*, 23(11):1222–1239, 2001.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. ICCV*, pages 886–893, 2005.

[4] J. Elder and S. Zucker. Computing contour closure. In *Proc. ECCV*, pages 399–412, 1996.

[5] D. Gavrila. Pedestrian detection from a moving vehicle. In *Proc. ECCV*, pages 37–49, 2000.

[6] M. Kumar, P. Torr, and A. Zisserman. Objcut. In *Proc. CVPR*, pages 18–25, 2005.

[7] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *Proc. CVPR*, pages 878–885, 2005.

[8] A. Opelt, A. Pinz, and A. Zisserman. A boundary-fragment-model for object detection. In *Proc. ECCV*, 2006.

[9] J. Shotton, A. Blake, and R. Cipolla. Contour-based learning for object detection. In *Proc. ICCV*, 2005.

[10] Z. Tu et al. Image parsing: unifying segmentation, detection, and recognition. In *Proc. ICCV*, page 18. IEEE, 2003.

[11] P. Viola et al. Detecting pedestrians using patterns of motion and appearance. In *Proc. ICCV*, pages 734–741, 2003.