The International Journal of Robotics Research OnlineFirst, published on July 21, 2009 as doi:10.1177/0278364909341884

Max Bajracharya
Baback Moghaddam
Andrew Howard
Shane Brennan
Larry H. Matthies

Jet Propulsion Laboratory,
California Institute of Technology,
Pasadena, CA 91109,
USA
maxb@robotics.jpl.nasa.gov

A Fast Stereo-based System for Detecting and Tracking Pedestrians from a Moving Vehicle

Abstract

In this paper we describe a fully integrated system for detecting, localizing, and tracking pedestrians from a moving vehicle. The system can reliably detect upright pedestrians to a range of 40 m in lightly cluttered urban environments. The system uses range data from stereo vision to segment the scene into regions of interest, from which shape features are extracted and used to classify pedestrians. The regions are tracked using shape and appearance features. Tracking is used to temporally filter classifications to improve performance and to estimate the velocity of pedestrians for use in path planning. The end-to-end system runs at 5 Hz on 1,024 × 768 *imagery using a standard 2.4 GHz Intel Core 2 Quad processor, and has been integrated and tested on multiple ground vehicles and environments. We show performance on a diverse set of datasets with groundtruth in outdoor environments with varying degrees of pedestrian density and clutter. In highly cluttered urban environments, the detection rates are on a par with state-of-the-art but significantly slower systems.*

KEY WORDS—pedestrian detection, human detection, stereo, tracking

1. Introduction

The ability of autonomous vehicles to detect and predict the motion of pedestrians or personnel in their vicinity is critical to ensure that the vehicles operate safely around people. Unmanned ground vehicles (UGVs) being developed for military applications are large, heavy, and potentially fast-moving vehicles. One of the highest-priority issues in the development of these UGVs is that they do not injure people, either during the research and development phase or in deployed operations. Vehicles must be able to detect people in urban and cross-country environments, including flat, uneven, and multi-level terrain, with widely varying degrees of clutter, occlusion, and illumination (and ultimately for operating day or night, in all weather, and in the presence of atmospheric obscurants). To support high-speed driving, reliable detection to ranges of approximately 100 m are likely to be necessary. The ability to detect pedestrians from a moving vehicle in a cluttered, dynamic urban environments is also applicable to automatic driver-assistance systems or smaller autonomous robots navigating in environments such as a sidewalk or marketplace.

In this paper we describe a fully integrated system capable of reliably detecting, localizing, and tracking upright (stationary, walking, or running) human adults at 5 Hz out to a range of 40 m from a moving platform. Although not explicitly designed to handle partial occlusion, non-upright postures, or children, the system performs reasonably well in these situations. Our approach uses imagery and dense range data from stereo cameras for the detection, tracking, and velocity estimation of pedestrians. The system runs on a standard 2.4 GHz Intel Core 2 Quad processor on 1,024 × 768 imagery. The ability to process this high-resolution imagery enables the system to achieve better performance at long range compared with other state-of-the-art implementations. As the system segments and classifies people based on stereo range data, it is largely invariant to the variability of pedestrians' appearance (due to the different types and styles of clothing) and scale. The system also handles different viewpoints (frontal versus side views) and poses (including articulations and walking) of pedestrians, and is robust to objects being carried or worn by them. Furthermore, the system makes no assumption of a ground-

The International Journal of Robotics Research
Vol. 00, No. 00, 2009, pp. 000–000
DOI: 10.1177/0278364909341884

Figures 1–11 appear in color online: http://ijr.sagepub.com

Fig. 1. Examples of test scenarios and the output of our pedestrian detection system (yellow boxes are detections with range and track ID text and a green overlay of the segmented person; the cyan boxes are missed detections).

plane to detect or track people, and similarly makes no assumption about the predictability of a person's motion other than a maximum velocity. When a vehicle motion estimate is not available from other sensors (such as an inertial navigation system (INS)), the system is also capable of visually estimating the motion of the vehicle, even in highly cluttered, dynamic scenes. However, the system does not require motion of the vehicle or people for detection.

The use of stereo vision is a key advantage of our approach. Research to date has not achieved the detection ranges or reliability needed in deployed systems to detect upright pedestrians in flat, relatively uncluttered terrain, let alone in more complex environments and with people in postures that are more difficult to detect. Range data is essential to solve this problem. Combining range data with high-resolution imagery may enable higher performance than range data alone because image appearance can complement shape information in range data and because cameras may offer higher angular resolution than typical range sensors. The experiments shown in Section 4.1 indicate that pixels-on-target is the key factor in the correct classification of people. This makes stereo vision a promising approach for several reasons: image resolution is high and will continue to increase, the physical size and power dissipation of the cameras and computers will continue to decrease, and stereo cameras provide range data and imagery that are automatically spatially and temporally registered. Our results show that a stereo-based approach is currently competitive with alternative sensors and can be improved with higher-resolution cameras. It can also be applied to infrared stereo imagery for low-light or night-time operations.

The novelty of our system resides primarily in the method of finding regions-of-interest from stereo data and the use of a small set of simple, computationally efficient shape features for classification, both of which are effective at ranges significantly further than most other systems have addressed. The classifier and tracker are both implementations of standard concepts. The system is also one of very few that has been tested and analyzed on a very large and diverse corpus of data.

The performance of the system is demonstrated on a variety of ground-truthed datasets in various outdoor environments, with different degrees of person density and clutter. An example of these scenes is shown in Figure 1. The majority of new datasets taken to evaluate the system consist of scenarios simulating the operation of a UGV traveling at moderate speed in semi-urban terrain (paved roads with light clutter and people walking along or into the road). In these scenarios, the system is capable of initial detections of pedestrians up to 60 m, and reliable detection and tracking of pedestrians up to 40 m. In addition to testing on ground-truthed datasets, we describe previous and upcoming live testing and evaluation of the fully integrated system running onboard a UGV in these scenarios. Finally, we present performance results of our system on recently published datasets of crowded street scenes. Although not specifically designed for highly cluttered urban environments, we show that results of our system are comparable

to the state-of-the-art systems while able to run significantly faster.

2. Related Work

There has been extensive research on pedestrian detection from manned and unmanned ground vehicles using scanning laser rangefinders (LIDAR) and monocular and stereo vision in visible, near-infrared, and thermal infrared wavelengths. Most such work assumes that the scene contains a dominant ground plane that supports all of the pedestrians in upright postures. Maximum detection ranges tend to be 30 m or less. Rates of missed detections and false alarms are not good enough to be satisfactory in deployed systems. Most prior work on pedestrian detection has been done for applications to smart automobiles, robotic vehicles, or surveillance. This literature is very large, so we only cover recent highlights and the main trends here.

2.1. Smart Automobiles

Research on pedestrian detection for smart automobiles has employed monocular vision (Shashua et al. 2004; Arndt et al. 2007; Ma et al. 2007) stereo vision (Sotelo et al. 2006; Bertozzi et al. 2007; Gavrila and Munder 2007; Liebe et al. 2007; Tomiuc et al. 2007), and LIDAR (Fuerstenberg et al. 2002). Vision-based methods have used visible (Shashua et al. 2004; Ma et al. 2007), near-infrared (Arndt et al. 2007), and thermal imagery (Bertozzi et al. 2007). Most work in this area has been strongly motivated by the requirement to be very low cost in eventual production.

The monocular vision work reported by Shashua et al. (2004) appears to be among the most mature in the automotive arena. They detect regions of interest (ROIs) in each image using a flat ground assumption to constrain the search, then extract gradient-based features from each ROI and classify and track the ROIs over successive frames. The range to objects is estimated by assuming that the bottom of each ROI is on the ground plane. The system uses 640 × 480 imagery with a 47° field of view and is designed to detect pedestrians at 10 Hz within 25 m of the camera. Single-frame classification performance evaluated with many hours of imagery recorded in urban driving was given as a false positive rate per ROI of 8% ("false positives per window", FPPW) at a probability of detection (Pd) of 93.5%. They process an average of 75 ROIs per image, which results in the system producing approximately six false alarms per image. Tracking is done over a minimum of four frames before results are output; multi-frame analysis reduces the false alarm rate by factors of between 10^3 and 10^6, depending on where the pedestrians appear and if/how they are moving. For the hardest case of stationary, out-of-path pedestrians, they reported a system-level Pd of 85% with 1.7 false positives per minute. Performance evaluation did not include partially occluded pedestrians.

Methods using stereo vision have a similar architecture, but use the range data to aid in detecting ROIs and to estimate the range to objects. The stereo vision systems generally output sparse depth maps with range to edge features; the best described systems use 320 × 240 imagery (Sotelo et al. 2006; Gavrila and Munder 2007) and also aim for a maximum range of 25 m. Details of the ROI detection, feature extraction, classification, and tracking algorithms vary by author. A key feature they have in common is that, although the range data from stereo is used in detecting ROIs, feature extraction and classification is done with image data, not range data. Gavrila and Munder (2007) reports frame-level performance at a Pd of 61% with 17.3 false positives per minute for pedestrians within ±4 m to each side of the vehicle path. This false positive rate is equivalent to a precision of 52.6%; precision is the fraction of reported detections that are really pedestrians. For trajectory-level performance, the false positive rate drops to 3.5 per minute. Sotelo et al. (2006) reports a Pd of 93.2% with a precision of 92.6%; since this is not evaluated on the same data set, it is unclear what explains the performance difference between these two systems. These two systems process imagery at 6–20 Hz with one 2.4 GHz Pentium 4 PC. Liebe's system (Liebe et al. 2007) also uses 320 × 240 imagery, but runs much more slowly. Their performance evaluation included pedestrians up to 50 m away with up to 70% occlusion; at a Pd of 42%, they experience 1.7 false alarms per frame. Presumably this lower performance is due at least in part to the greater maximum range and partial occlusions in the test data. Extensions of this work include that of Ess et al. (2007, 2008), which uses 640 × 480 imagery and reports a Pd of 40% to 55% at one false positive per frame on cluttered urban sidewalks. We specifically compare our system directly to theirs (in Section 4.2) as they have published their datasets.

Near-infrared and thermal infrared imagery have been employed to address operation at night (Arndt et al. 2007; Bertozzi et al. 2007). The algorithm architectures are analogous to those above. Work with LIDAR for the automotive domain includes use of the four-beam scanner by IBEO (Fuerstenberg et al. 2002), which now has a range exceeding 100 m. Claims are made for very good pedestrian detection and false alarm rates, but the systems and experiments are described in less detail than other related work, making performance hard to compare.

2.2. Robotic Vehicles

Most work on pedestrian detection for robotic vehicles in outdoor applications is being done under the Army Research Lab (ARL) Robotics Collaborative Technology Alliance (RCTA) program. This work includes methods that perform range sensing with two-dimensional LIDAR, three-dimensional LIDAR,

stereo vision, and/or structure from motion and do image sensing with visible and/or thermal infrared cameras. At a high level, algorithm architectures are analogous to the systems for the automotive domain, involving ROI detection, classification, and tracking, although the order and details of these steps differ. Some approaches (Navarro-Serment et al. 2008; Thornton et al. 2008) detect which objects are moving before performing classification. As a group, there is more emphasis in this domain on classification based on the three-dimensional shape of the objects as perceived by LIDAR or stereo vision than there is in the automotive domain. The feature extraction and classification algorithms tend to be simpler than those used in either the automotive or video surveillance domains.

Thornton et al. (2008) uses a LIDAR that scans 180° horizontally and has many beams vertically to provide a three-dimensional range image; sensor details are proprietary, but the functionality is similar to the commercially available LIDAR from Velodyne. Above-ground objects are segmented into distinct point clouds, which are tracked to estimate their velocity. A "strength-of-detection" function combines simple features of the density, shape, velocity, and temporal stability of the point cloud to provide a confidence measure that the point cloud is a pedestrian. Preliminary work was also done with long-wave thermal infrared imagery to detect pedestrians beyond the range of the LIDAR and in non-upright postures that are hard to recognize with LIDAR data. Navarro-Serment et al. (2008) employs a similar sequence of operations with two-dimensional LIDAR scans in a plane parallel to the ground.

Stereo vision-based approaches have been explored in the RCTA program by Howard et al. (2007) and Bajracharya et al. (2008) at the Jet Propulsion Laboratory (JPL), in an earlier version of the work reported here, and by Abd-Almageed et al. (2007) at the University of Maryland (UMd). Howard processed $1{,}024 \times 768$ stereo imagery into 512×384 range images (60° field of view), transformed the range data into two-dimensional maps in a horizontal reference frame, segmented upright objects in those maps, and performed classification on the resulting three-dimensional point clouds for each object. This was based on a dense, area correlation-based stereo vision algorithm that outputs range estimates for most pixels of the image; this is distinct from the main trend in automotive applications, which use sparse range data at the edges. The point clouds were also used to compute rectangular ROIs in image space for input to an image-based classifier. The system ran at 3.75 Hz. Bajracharya et al. (2008) extended this approach by improving detection of candidate objects with the range data, improving the feature extraction and shape-based classification stages of the system, and modifying the system to run on $1{,}024 \times 768$ imagery at the same rate as its predecessor. Abd-Almageed et al. (2007) used image ROIs computed from the JPL stereo vision-based range data as input to a classifier based on Adaboost.

The RCTA program conducts "Safe Operations" (SafeOps) field experiments in the fall of each year to quantitatively measure the performance of pedestrian detection systems. All of the systems discussed above were evaluated in the FY2007 experiment, which was on a flat road about 250 m long with 10 moving pedestrians, four stationary mannequins, and assorted moving and stationary clutter objects; overall the scene was relatively uncluttered. For the FY2007 experiment, the course was run 32 times to generate performance statistics for LIDAR and stereo vision-based systems; results are discussed by Bodt (2008). The median distance to first detection of people and mannequins varied from about 25 to 45 m for LIDAR-based systems and 25 to 32 m for stereo vision-based systems. Detection rates were evaluated as a function of how many frames each target was detected in on a given run, which we will call "persistence". For a persistence of four frames, algorithms using three-dimensional LIDAR data had detection rates of 95–100% for moving people and stationary mannequins combined. The detection rates for stereo vision-based algorithms are ambiguous, because the evaluation may not (yet) have properly scored targets that were not in the field of view of the cameras. With that caveat, the four frame persistence for stereo vision was at least 57%. Classification errors on clutter objects were scored similarly; for four frame persistence, 10–20% of clutter objects were misclassified as human for the algorithms using three-dimensional LIDAR or stereo vision (precision of 80–90%). Pickup trucks and human-sized crates, in particular, caused classification errors. The reasons for this have not yet been analyzed in depth, but it may be that for these sensors the range data on pickup trucks breaks up into human-sized blobs. Results available to date from the experiment analysis do not allow direct comparison of the false alarm rate from the SafeOps experiment to false alarm rates published in the automotive domain; moreover, the types of scenes in the respective data sets differ enough that such a comparison would be inconclusive.

2.3. Surveillance

Work on pedestrian detection in the surveillance arena largely divides into work with image sequences from stationary cameras, where background subtraction and/or image differencing is used to detect moving objects (Beymer and Konolige 1999; Viola et al. 2003; Zhao et al. 2008), and work that applies trained pattern classifiers to individual images (Dalal and Triggs 2005; Sabzmeydani and Mori 2007; Seeman et al. 2007; Tuzel et al. 2007; Wu and Nevatia 2007). The former group is less relevant here, because background subtraction and temporal image differencing are more difficult to use from moving cameras. Stereo data has been used in this area to segment, classify, and track people (Beymer and Konolige 1999), however the methods still rely on background modeling (Eveland et al. 1998) and so do not handle camera translations, and have

been limited in range (to less than 20 m). The latter group uses a variety of feature extraction and classification methods to achieve better Pd and FPPW rates than single-frame results reported in the automotive pedestrian detection literature; however, the results are not directly comparable for a number of reasons. Since real-time performance on embedded computers is not required, computational requirements generally are higher or not stated. The testing protocol often used is not a good match to driving scenarios, since it either uses image databases where positive examples are already centered in image chips or performs exhaustive search over position and scale of ROIs in test imagery. Finally, not having a tracking module that helps detection and false alarm performance leads to different algorithm design and computational tradeoffs. For these reasons we do not elaborate on these methods here; nevertheless, this research does offer the potential to improve single-frame performance of classifiers used in automotive and robotic vehicle domains.

3. System Description

Our pedestrian detection system is primarily designed to enable autonomous vehicles to safely navigate when people are present. Consequently, the system must be able to detect a person with enough time for a planner to generate a plan to avoid the person and the vehicle to execute this plan. Furthermore, it must be able to predict the person's motion and maintain a false positive rate that prevents the vehicle from unnecessarily avoiding objects. The detection system's requirements are highly dependent on the overall system configuration and requirements, however we are specifically targeting a car-sized vehicle driving at 30 km h^{-1} in lightly cluttered terrain, and ultimately desire to drive at 50 km h^{-1} in highly cluttered terrain.

Our system consists of the following modules, which are each described in more detail in the balance of this section.

- *Stereo vision*. The stereo vision module takes synchronized images from a pair of cameras and computes a dense range image.
- *Visual odometry*. The visual odometry module takes two sequential pairs of stereo images and computes the frame-to-frame camera motion. In practice, if a good pose estimate is available from other vehicle sensors (such as an INS), this step is skipped.
- *ROI detection*. The ROI detection module projects stereo data into a polar-perspective map (PPM) and then segments the map to produce clusters of pixels corresponding to upright objects. These clusters are then filtered for human-sized objects based on their three-dimensional shape statistics.
- *Classification*. The classification module computes geometric features of the 3D point cloud of each ROI and classifies the object, resulting in a probability of being human.
- *Tracking*. The tracking module associates ROIs in sequential frames, accounting for vehicle motion, and estimates the velocity of the detected objects. The probabilities for each tracked object are filtered over time to produce a final detection result.

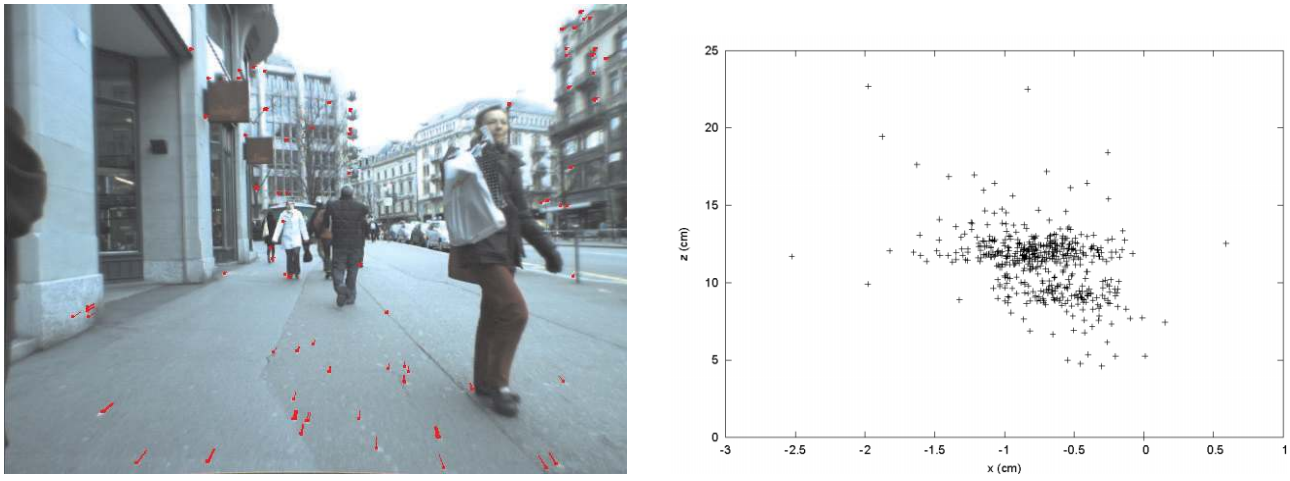
The system architecture allows the possibility of using appearance and motion features to improve the classification of people, but we currently do not make use of these features. We intend to use them in the future to improve the performance of the system, particularly on partially occluded or non-upright people. However, one advantage of only using shape information is that the algorithm could, in principle, be applied to range data from other sensors.

3.1. Stereo Vision

The first step in our system is to compute dense range data from stereo images. We use a multi-processor version of the algorithm described by Goldberg et al. (2002) previously used on the NASA Mars Exploration Rovers and in the DARPA PerceptOR program. On a 2.4 GHz Intel Core 2 Quad processor, the algorithm can process 1,024 × 768 imagery at 10 frames per second. The algorithm has also been ported to a filed-programmable gate array (FPGA), which can process 1,024 × 768 imagery at 15 frames per second. When run in software, the stereo processing dominates the computation time of the overall system and is the only component of the system that takes advantage of the multiple cores of the CPU.

3.2. Visual Odometry

When the pose of the vehicle is not available from an INS, accurate knowledge of frame-to-frame camera motion is produced by visual pose estimation. We use the visual odometry algorithm described by Howard (2008) that tracks point features in imagery and uses the dense range data to provide the range to each feature. Briefly, the algorithm detects features in each frame with a corner detector, matches features between frames using their sum-of-absolute-differences over local windows, finds the largest set of self-consistent matches (inliers), and then finds the frame-to-frame motion that minimizes the reprojection error for features in the inlier set. The algorithm exploits intermediate steps in the stereo processing pipeline to optimize execution and is able to process 1,024 × 768 images in 10 to 20 ms per frame. In static environments, typical accuracy is better than 1 m over 400 m of travel.



(a) (b)

Fig. 2. (a) An example image from an urban sequence (Ess et al. 2007), with feature tracks from visual odometry. (b) Frame-to-frame translations computed by visual odometry, in centimeters; the x- and z-axes correspond to lateral and forward motion of the camera, respectively

This algorithm also performs reliably in cluttered, dynamic environments (such as urban sidewalks). For example, Figure 2(a) shows an image from one of the urban sequences described in Ess et al. (2007). While there is no ground truth for this sequence, we know from visual inspection that the camera motion is smooth and approximately linear; we can therefore assess the reliability of visual odometry by looking at the estimated frame-to-frame change in pose. The scatter plot in Figure 2(b) shows the camera translations in forward and lateral directions. Note that there are no large jumps or kinematically infeasible lateral translations, indicating that visual odometry has correctly extracted the camera motion while ignoring the independent movers. Visual odometry cannot work for all scenes, however; if the environment is heavily occluded by movers additional sensors or kinematic constraints must be applied to disambiguate the multiple motions present in the scene.

3.3. ROI Detection

Detecting ROI areas from the stereo data serves as a focus-of-attention mechanism to reduce the runtime of subsequent classifiers and segments foreground pixels from background pixels in a region. This allows a shape-based classifier to be run on the 3D points that make up a specific object, rather than sliding a window over the image and explicitly performing foreground/background segmentation in each window.

The steps of the ROI detection algorithm are illustrated in Figure 3. Figure 3(a) shows a simple test scene with two people at 5 and 30 m distances from the cameras. Figure 3(b) shows a depth map produced by the dense stereo matching algorithm; color codes represent the distance, with red closest, blue furthest, and dark red representing pixels with no range data. The range data is projected into a two-dimensional grid map, which is then segmented based on map cell statistics. In order to capture the variable resolution and preserve the coherency of the stereo range data, the map is represented as a PPM. Unlike a traditional Cartesian map, which is divided into cells of fixed size in Cartesian (x, y) space, the PPM is divided into cells with a fixed angular resolution but variable range resolution in polar (r, θ) space. The range resolution (r-axis, up each column in Figure 3(c)) corresponds to stereo disparity, proportional to inverse range, and consequently accounts for stereo range error by accumulating all of the points that lie within the expected stereo range error. Each row in the PPM corresponds to a fixed interval of stereo disparity; each column corresponds to one (or more) columns of the depth map. The stereo range data is transformed into a gravity-leveled frame and then projected into the PPM, which accumulates the number of points projected into each cell. The map is then smoothed with an averaging filter with an adaptive bandwidth in polar space corresponding to a fixed bandwidth in Cartesian space. For computational efficiency the filter is implemented by first computing the integral image of the map. Figure 3(c) shows the PPM for the depth map in Figure 3(b) after smoothing. The diagonal row of blobs on the left corresponds to the row of trees. The person at 5 m is the distinct blob at the bottom of the map. About halfway up the image, the blob to the right of the trees is from the overhanging branch visible at the top of the image in Figure 3(a). Farther up the image, another blob on the right side of the trees corresponds to the person at 30 m.

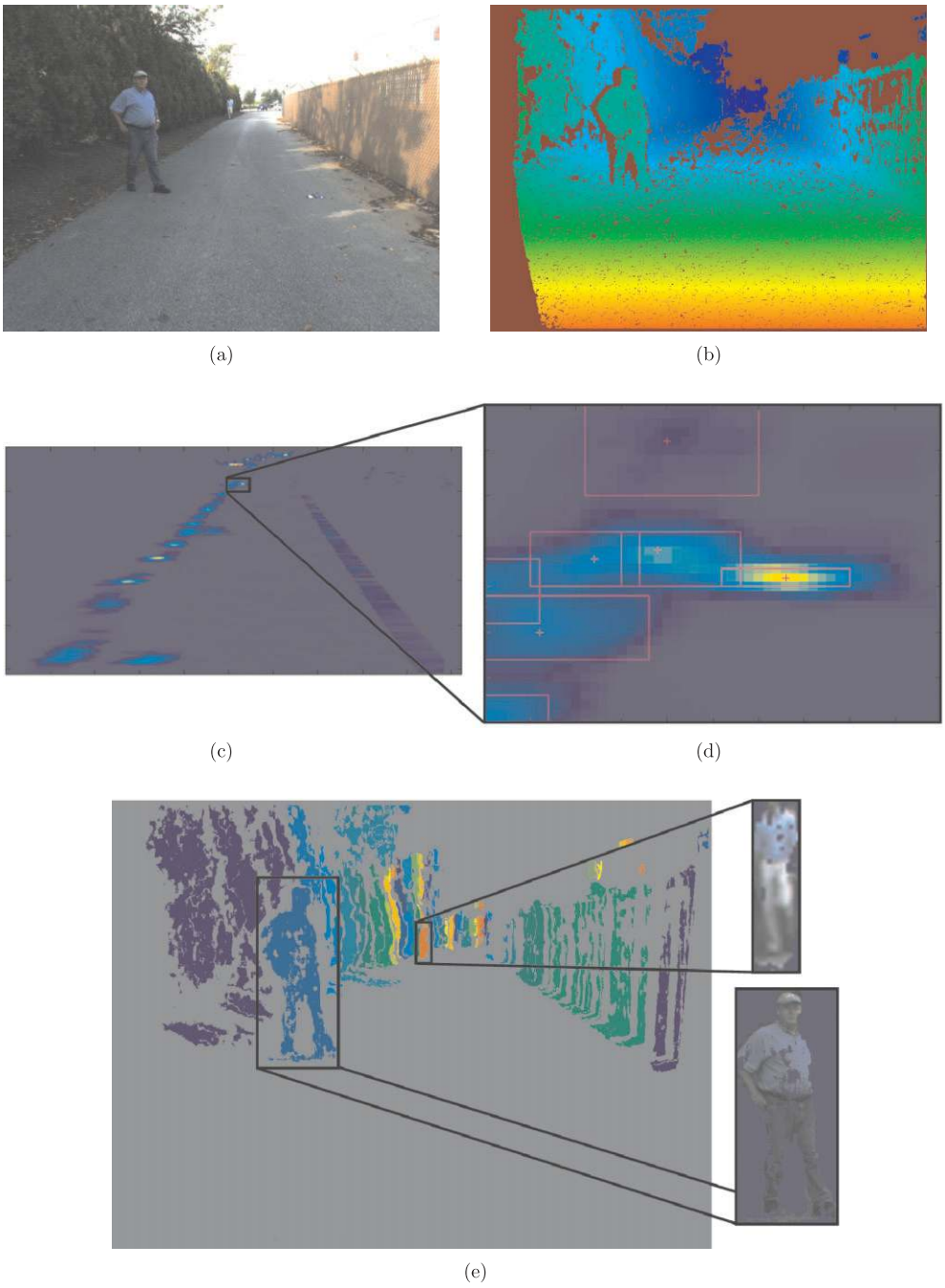


Fig. 3. An example of the stereo-based segmentation for region-of-interest detection: (a) the left image of a stereo pair; (b) the resulting depth map; (c) the PPM of point counts smoothed with an averaging filter; (d) a close up of the map with segmented regions overlaid; and (e) the segmented regions, with examples of the foreground/background separation.

Overhangs are currently not removed before projecting data into the PPM, however segmented blobs are post-processed to remove outliers. The additional clutter in the PPM caused by overhangs is generally insignificant in the semi-urban datasets used, but can be problematic in urban environments, resulting in missed detections. This will be a subject of future work.

After smoothing, the map gradient is used to find all of the peaks in the map. The peaks are grown down to valleys (an inflection point in the gradient), resulting in a segmentation of

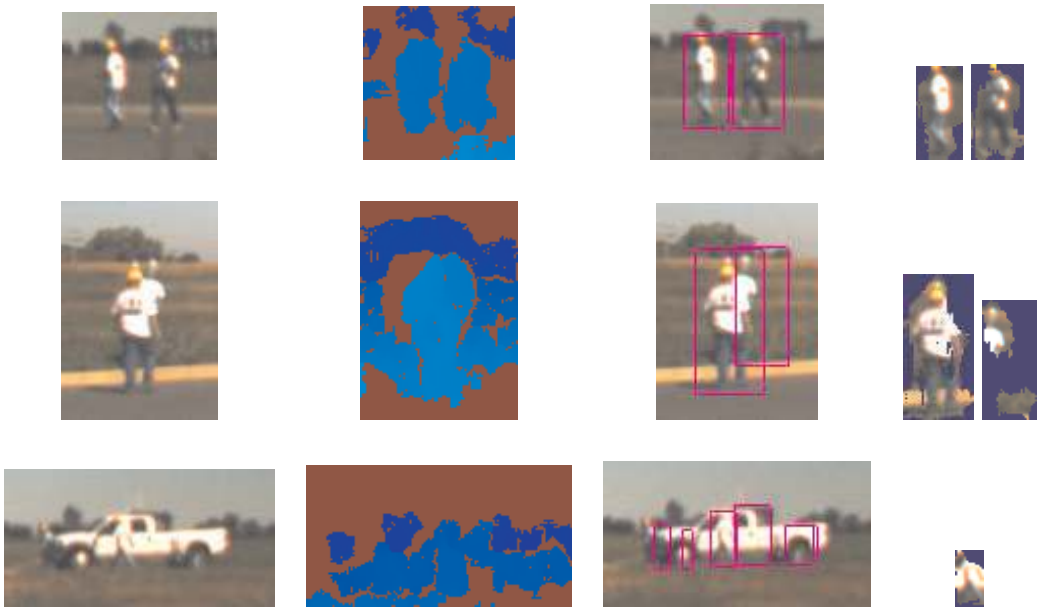


Fig. 4. Examples of ROI detection from a semi-urban sequence. The first column shows a thumbnail of the image; the second shows the depth map; the third column shows the segmented regions; and the fourth column shows the foreground/background segmentation for specific regions. The top row shows an easily separable case with people at 30 m; the middle row shows a case of partial occlusion, with a person at 30 m; and the bottom row shows a person at 45 m who is difficult to distinguish visually.

the map. As the minimum expected size of the objects being detecting is known, segmented blobs whose peaks fall within half of this size are then merged together. Figure 3(d) gives a close-up of the PPM around the person at 30 m; although the map blob corresponding to the person is not completely disjoint from the blob for the nearest tree, it is still segmented as a separate ROI. Figure 3(e) illustrates the segmentation results in image space.

Figure 4 shows several more examples of segmentation, including more challenging cases. The top row shows the segmentation of two people at approximately 30 m, where the people are easily distinguishable in the stereo data. Stereo matching causes "foreground fattening" of the regions containing people, but the effect tends to be consistent and so can be accounted for during classification. Alternatively, a more sophisticated stereo algorithm could be applied in each region to reduce this effect, but we have not yet implemented this. The middle rows shows the segmentation of a person at 25 m partially occluding a mannequin several meters behind them. As the mannequin falls into a cell well behind the person, the mannequin is segmented correctly, but includes a portion of the ground. The bottom row shows a person at 45 m walking in front of a vehicle with a similar color. In this case, the person is difficult to distinguish from the vehicle visually, but can still be segmented correctly because of the range data on the person at other locations on his body. Note, however, that the vehicle is over-segmented into many separate regions. This is due to the patchy stereo of the flat vehicle, resulting in many regions that are then smoothed and merged, resulting in human-sized regions. Overall, on this semi-urban data, our approach rarely fails to correctly segment a person closer than 60 m. Even on the urban datasets, the segmentation rarely fails to detect people when the stereo coverage is sufficient. The problems with the segmentation tend to be with over-segmenting non-human objects into human-sized objects or merging multiple people into a single region. The latter could be addressed by improving the post-processing of the regions, or by selecting sub-regions to provide to the classifier. The former is more difficult to address, but could potentially be alleviated by using multiply sized and oriented filters, prefiltering the range data before projecting it into the PPM, or improving the stereo algorithm to provide more range data in low-texture areas of the image.

3.4. Classification

Geometric features of each segmented three-dimensional point cloud are used to classify them as human or not human based on shape. For efficiency, the regions are first prefiltered, and shape-based features are then computed on the remaining regions. The regions are classified using a discriminative quadratic classifier based on a logistic regression model.

Prefiltering of regions based on shape moments is used to reduce the number of regions and create more balanced training data. The prefilter uses a fixed threshold on the width, height, and depth variance of each segmented region. This threshold is simply selected as the 3σ values obtained from the training data. After prefiltering, the features used for classification are computed for each region's point cloud.

Our features include the fixed-frame shape moments (variances of point clouds in a fixed frame), rotationally invariant shape moments (the eigenvalues of the point cloud's scatter matrix), and "soft-counts" of various width, height, depth, and volume constraints. The logarithmic and empirical logit transforms of these moments and counts are used to improve the normality of the feature distribution (resulting in "soft" counts, as opposed to raw counts). The features were selected based on our prior experience and similar features shown to be effective in other work (Howard et al. 2007; Thornton et al. 2008).

To compute the features, we start by centering the point cloud about the x-axis by its mean value and setting the minimum depth z and height y to zero. The first feature is defined by the logarithm of the second-order moment of the height:

$$f_1 = -\log(\sigma_y^2). \tag{1}$$

We use the negative sign for the logs in order to have feature values be more positive for the (smaller) human blobs. We also off-center the y moment by redefining it as $\sigma_y^2 = E(y - 0.5)^2$ where E denotes the expectation operator. The particular offset value (of 0.5 m) was experimentally found to enhance performance and more generally could be automatically learned from data. Finally, we center the distribution of all features by subtracting a constant shift value so the "cross-over" value of each feature is near zero. Such linear shifts in the log-domain correspond to (arbitrary) scale factors in the original coordinates and are omitted in the equations presented.

The "soft-count" features are defined by the number of points that fall inside certain preset coordinate bounds (or volumes). Such count-based features ignore "true shape" and focus instead on the object's size or extent. Unlike moment-based features, count-based features are more tolerant of outlier noise and some artifacts of stereo processing. Naturally there are strong correlations between these two different sets of features. However, this correlation or redundancy can be quite helpful for modeling purposes. For the total number of points n in a blob point cloud, we define $n_x = \#(|x| < 1)$ as the number (subset) of three-dimensional points whose x value is less than 1 m (in absolute value), $n_{y_0} = \#(y < 2)$ and $n_{y_1} = \#(y > 1)$ as the number of points whose height value is less than 2 m and greater than 1 m, and $n_{z_0} = \#(z < 4)$ and $n_{z_1} = \#(z < 3.5)$ as the number of points with a depth value less than 4 and 3.5 m, respectively. We also define n_v to be the number of three-dimensional points that satisfy all three width, height, and depth constraints simultaneously (i.e. the number of points that fall within the prescribed rectangular volume of size 1 m × 2 m × 4 m). Although these constraints were selected empirically, the process could easily be automated. In order to normalize the data as well as account for uncertainty due to the sample size (n), we use a logit transform with an empirical prior count c:

$$\begin{aligned}
f_2 &= \log \frac{n_x + c_x}{n - n_x + c_x}, \\
f_3 &= \log \frac{n_{y_0} + c_{y_0}}{n - n_{y_0} + c_{y_0}}, \\
f_4 &= \log \frac{n_{z_0} + c_{z_0}}{n - n_{z_0} + c_{z_0}}, \\
f_5 &= \log \frac{n_v + c_v}{n - n_v + c_v}, \\
f_6 &= \log \frac{n_{y_1} + c_{y_1}}{n - n_{y_1} + c_{y_1}}, \\
f_7 &= \log \frac{n_{z_1} + c_{z_1}}{n - n_{z_1} + c_{z_1}}.
\end{aligned} \tag{2}$$

The rotationally invariant features are the logarithms of the eigenvalues of the point cloud's covariance (inertia) matrix, where $(\lambda_x, \lambda_y, \lambda_z)$ correspond to the major, intermediate, and minor axes, respectively:

$$\begin{aligned}
f_8 &= -\log(\lambda_x), \\
f_9 &= -\log(\lambda_y), \\
f_{10} &= -\log(\lambda_z).
\end{aligned} \tag{3}$$

We note that f_8 would be redundant with f_1 if all of the blobs were oriented correctly (upright and "facing" downrange). However, this is often not the case due to artifacts in stereo processing or slight errors in roll/pitch estimates, resulting in point clouds that are tilted and/or slanted. We once again use the negative sign convention (so human feature values are more positive) and likewise use appropriate additive shifts to center the distributions even though these values are not shown in the equations.

Analysis of the shape features indicated that a linear classifier (with a linear decision boundary) was too simple to always work effectively. However, a more complex decision boundary can be achieved while still using a linear classifier (which is desirable for its computational efficiency and robustness) by expanding the feature set to use higher-order terms. Specifically, a quadratic decision boundary is modeled using the augmented feature set:

$$\mathbf{x} = [1 \quad \{f_i\} \quad \{f_i f_j\}_{i<j} \quad \{f_i^2\}]^{\mathrm{T}}. \tag{4}$$

Using this feature vector, we use Bayesian parameter estimation for a discriminative classifier based on a standard

generalized linear model for binary outcomes (human versus non-human). For this probabilistic model, the *logit* of the class membership probability $p(y = +1)$ is modeled by the linear term $\mathbf{w}^{\mathrm{T}}\mathbf{x}$, where $\mathbf{w}$ is the vector of feature weights (our classifier parameters). Equivalently, this means using a *logistic* sigmoid function on $\mathbf{w}^{\mathrm{T}}\mathbf{x}$ to model output probabilities $p \in [0, 1]$:

$$p(y_i = +1 \mid \mathbf{x}_i, \mathbf{w}) = \mathtt{logit}^{-1}(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i) = \frac{1}{1 + e^{-\mathbf{w}^{\mathrm{T}}\mathbf{x}_i}}. \quad (5)$$

This is simply the probability for a Bernoulli model (of being human) given an input feature vector $\mathbf{x}_i$. Given that our y labels are ± 1 and that $p(y = +1) = 1 - p(y = -1)$ and exploiting the symmetry of the logistic function itself, the full likelihood for the entire training set can be written in this compact form

$$p(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = \prod_{i=1}^{n} (1 + e^{-y_i \mathbf{w}^{\mathrm{T}}\mathbf{x}_i})^{-1}, \quad (6)$$

where $\mathbf{y}$ is the vector of output labels and the matrix $\mathbf{X}$ collects all training feature vectors $\mathbf{x}_i$ in its columns.

All that remains is to posit a prior distribution on our parameters, and for convenience we use a zero-mean Gaussian $p(\mathbf{w}) = \mathcal{N}(0, \Sigma)$. If this was the first training set we encountered, we could use a non-informative (diffuse) prior by setting Σ very large (and diagonal). More importantly, we can use the posterior distribution inferred from previous training sets as the prior distribution on new sets. The Bayesian derivation is completed by examining the posterior distribution which is proportional to the joint distribution

$$p(\mathbf{w} \mid \mathbf{X}, \mathbf{y}) \propto p(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) p(\mathbf{w}). \quad (7)$$

More conveniently, we form the log-posterior

$$\Psi(\mathbf{w}) = \log p(\mathbf{w} \mid \mathbf{X}, \mathbf{y}) = -\frac{1}{2}\mathbf{w}^{\mathrm{T}}\Sigma^{-1}\mathbf{w} - \sum_{i=1}^{n} \log(1 + e^{-y_i \mathbf{w}^{\mathrm{T}}\mathbf{x}_i}) + \text{constant}, \quad (8)$$

which for this model (using a log-quadratic prior) is conveniently log-concave. This means that the posterior distribution has a single global maximum which is easy to find by iterative non-linear optimization methods. The technique of choice for this class of models is the iteratively reweighted least squares (IRLS) algorithm, which uses Newton–Raphson updates to solve a set n coupled non-linear equations for $\nabla\Psi(\mathbf{w}) = 0$. Having reached the mode $\hat{\mathbf{w}}$ this optimization procedure also yields the local curvature or Hessian: $\mathbf{H} = \nabla^2\Psi(\mathbf{w})$. Although the log-posterior is unimodal, it is generally skewed (non-Gaussian) but if n is large, then a Gaussian approximation becomes increasingly accurate. Therefore, it is often adequate to model the $\mathbf{w}$ posterior by a Gaussian with mean $\hat{\mathbf{w}}$ and covariance matrix defined by the (negative) inverse Hessian, $\mathbf{V} = -\mathbf{H}^{-1}$

$$p(\mathbf{w} \mid \mathbf{X}, \mathbf{y}) \approx \mathcal{N}(\hat{\mathbf{w}}, \mathbf{V}). \quad (9)$$

We note that because we have an analytic closed-form expression for the (unnormalized) joint distribution of $(\mathbf{w}, y)$ it is not difficult to stochastically sample from the *exact* posterior of $\mathbf{w}$ using standard Markov chain Monte Carlo (MCMC) methods (e.g. the Metropolis–Hastings algorithm or importance resampling, etc.). When there are large number of training data, the simple Gaussian modal approximation in Equation (9) is usually sufficient for posterior predictive sampling (where it is trivially easy to sample from a multivariate Gaussian).

The *expected* posterior probability of being human for a new feature vector $\mathbf{x}_*$ is then given by marginalizing over the uncertainty in the $\mathbf{w}$ posterior

$$Ep(y_* = +1 \mid \mathbf{x}_*, \mathbf{X}, \mathbf{y}) = \int \frac{1}{1 + e^{-\mathbf{w}^{\mathrm{T}}\mathbf{x}_*}} p(\mathbf{w} \mid \mathbf{X}, \mathbf{y})\, d\mathbf{w}, \quad (10)$$

which can be reduced to a simple one-dimensional integral by working with the posterior distribution of the scalar random variable $\mathbf{w}^{\mathrm{T}}\mathbf{x}_*$. If instead of the Gaussian approximation to the posterior, we wish to use the exact posterior to evaluate this integral, stochastic methods using MCMC can be used. Nevertheless, it is convenient to output a single maximum *a posteriori* (MAP) estimate of the output probability by using the mode $\hat{\mathbf{w}}$

$$\hat{\pi}_* = \frac{1}{1 + e^{-\hat{\mathbf{w}}^{\mathrm{T}}\mathbf{x}_*}}. \quad (11)$$

This approximate predictive probability becomes more accurate as the number of observations n approaches infinity (since the posterior then approaches a δ function centered at $\hat{\mathbf{w}}$). We use the MAP estimate in all of the results presented in this paper.

3.5. Tracking

Tracking ROIs in the scene is used to both reduce incorrect detections and estimate the velocity of the detected objects. By associating ROIs across multiple frames, the single frame classifications can be aggregated to eliminate false positives. Similarly, using the positions of a tracked object from stereo and the motion of the vehicle, estimated by visual odometry or provided by an INS, the velocity of the object can be computed and extrapolated to provide a predicted motion to a path planner. The tracking algorithm is designed to be extremely computationally efficient and makes very few assumptions about the motions of objects.

Tracking of ROIs is actually implemented as data association, rather than explicit tracking. The ROIs extracted in a new frame are matched to existing nearby tracks by computing a cost based on each ROI's segmented foreground appearance

and then solving a one-to-one assignment problem. For computational efficiency and simplicity, the cost between a ROI and a track is computed by comparing the new ROI to the last ROI in the track. Only ROIs within a fixed distance are considered; the distance is computed by using an assumed maximum velocity of 2 m s^{-1} in any direction for each object. The cost between ROIs is then computed as the Bhattacharyya distance of a color (RGB) histogram between each ROI. The resulting linear assignment problem could be solved optimally by the Hungarian method or sub-optimally with a greedy algorithm, but these methods are relatively expensive ($O(n^3)$ and $O(n^2 \log n)$, respectively). Instead, we simply require assignments to be co-occurring minima ($O(n^2)$). If a ROI does not match an existing track, a new track is started. Tracks that are not matched for a fixed number of frames are eliminated. To reduce the number of ROIs tracked, only ROIs that pass the pre-filter based on size variances (Section 3.4) are considered. This also increases the stability of the ROIs by eliminating small, similarly colored regions nearby a larger region. Although we have experimented with many-to-one and one-to-many matching, we found one-to-one matching to be sufficient and simpler. We have not yet invested in more sophisticated methods, such as multiple hypothesis tracking, joint probabilistic data association filters, or kernel-based tracking, as the need has not justified the increased computational cost. An example of tracked pedestrians is shown in Figure 5. Figure 5(c) shows the individual tracks for the scene shown in Figure 5(b) (the multiple overlapping tracks are due to the fact that the vehicle, whose path is shown as the red line, doubled-backed across the intersection).

Analyzing the single frame output of our classifier for each track, we observed that many false positives were only present in a single frame or in multiple non-consecutive frames. Conversely, true positives (the pedestrians) were detected consistently over many frames, and when detections were missed it was typically for only one or two frames before the person was detected again. To eliminate the spikes in classification scores that led to false positives, while still maintaining detections on true positives where the classification score dropped for a small number of frames, we considered several methods of filtering the scores. These included computing the mean, median, maximum, and minimum score over a varying number of frames, and waiting a varying number of frames required to make a classification decision. In our experiments, two different combinations of the filtering method and minimum number of frames were found to work well. The first combination was to compute the median of three consecutive scores and requiring three consecutive frames of detection before making a classification decision. The second combination was to compute the minimum of four consecutive scores and require two consecutive frames of detection before making a classification decision. We ultimately fielded the first combination, but temporal filtering can also be disabled depending on the classifier operating point.

Trade-offs in temporally filtering the classification scores include the latency it introduces when declaring detections and the quality of tracking (length of tracks). Although temporal filtering can eliminate spurious detections, it also reduces the true positives. This results in the reduction of the detection rate at high false alarm per frame (FAPF) rates, but generally increases the detection rates at low FAPF rates, as shown in Figure 5(a).

The velocity of tracks is estimated by fitting a linear motion model over a sliding window of detections. We originally utilized independent Kalman filters to compute the expected position and velocity of each track, but found that due to the periodic motion of a person's gait, it did not provide significantly better results. Comparing the frame-to-frame position of a walking person tends to result in an oscillating velocity, but fitting a linear model over several frames smooths the motion. We estimate the position and velocity uncertainty by combining the expected stereo error with the model fit. An example of the computed velocity vectors and variances is shown in Figure 5(d) for the scene in Figure 5(b).

4. Experimental Results

The end-to-end system has been tested on datasets with hand-labeled ground-truth and integrated onboard a vehicle for live testing. The primary datasets were collected from the vehicle on which the system was integrated in semi-urban, lightly cluttered scenarios. Although relatively simple compared with what a deployed system might encounter, they are representative of the RCTA SafeOps field experiments used to evaluate the system. The results on the datasets show that our system can achieve initial detections at a range of 60 m, with detections reliable enough for autonomous navigation out to 40 m. To demonstrate that the system's performance is competitive with state-of-the-art systems in highly cluttered, urban scenarios, we also make use of datasets published by Ess et al. (2007, 2008). We show that we can achieve performance similar to Ess et al. on these datasets while running at 10 Hz.

4.1. Semi-urban Datasets

The primary datasets used to evaluate the system use input imagery from a three CCD color stereo camera pair with 1,024 × 768 pixels, a 50 cm baseline, a field of view approximately 60° wide, and with frame rates between 3.5 and 10 Hz. The cameras were either mounted on the roof of a sports utility vehicle (SUV) at a height of approximately 2 m above the ground, and pointed down by approximately 5°, or on the pan–tilt head of an unmanned vehicle at a height of approximate 2 m above the ground, and pointed down by 20°. The scenarios include the vehicle driving down a road at speeds varying from 15 to 30 km h^{-1}, with stationary mannequins and people

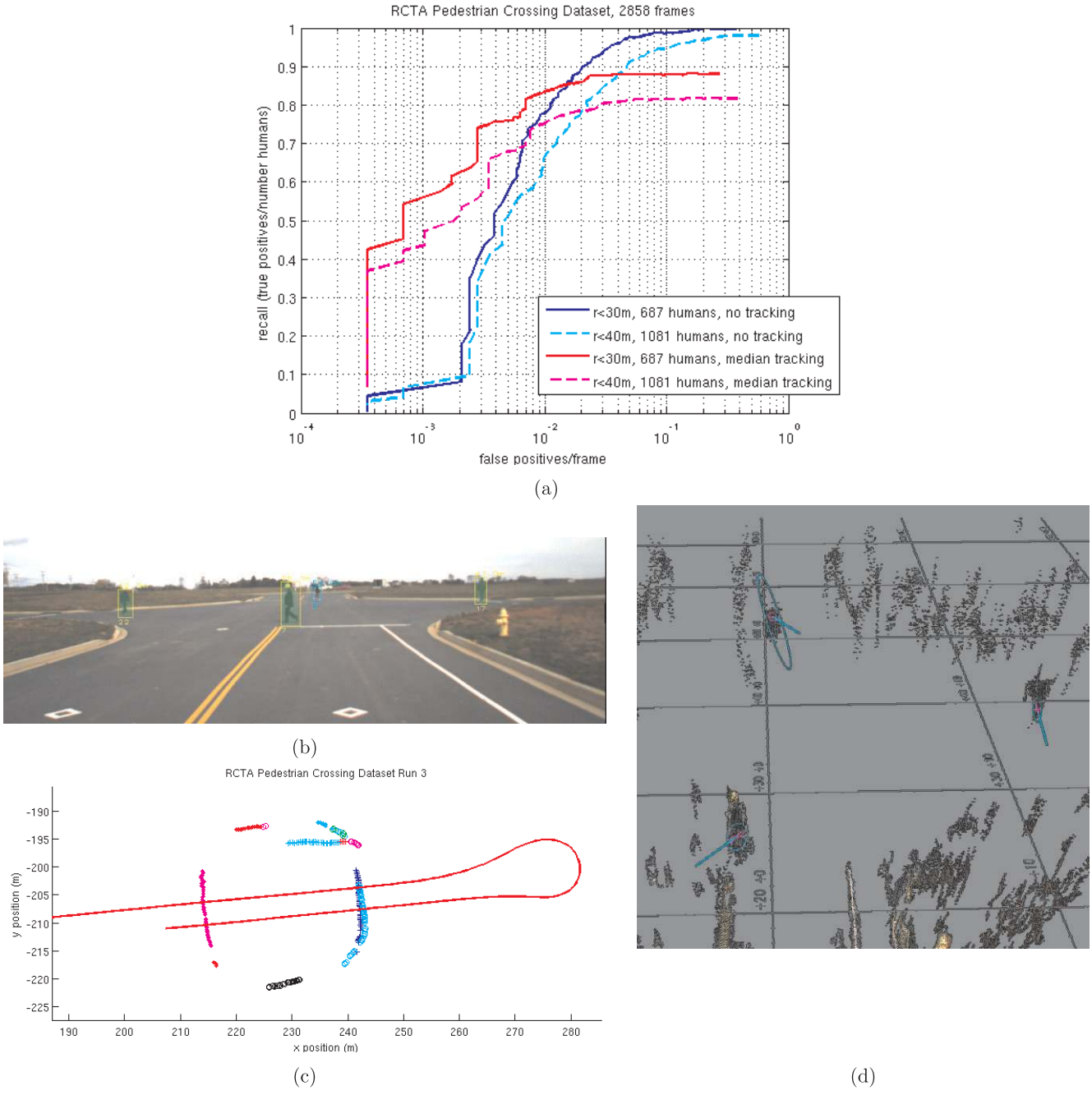


Fig. 5. (a) The performance for a single semi-urban sequence illustrating how tracking can be beneficial at low FAPF, but reduces the detection rate at high FAPF by introducing a latency. (b) An example detection from the scene (yellow boxes are detections, with a green overlay of the segmented person; the cyan boxes are missed detections). (c) The individual tracks detected during the run, with the vehicle path shown as the red line; note that the vehicle double-backed across the intersection, resulting in overlapping tracks. (d) The 3D point cloud of a region with estimated velocity vectors (cyan lines) and uncertainties (cyan ellipses) for the detections in (b).

standing, walking, and running along the side of and across the road in varying directions. The scene also contains stationary and moving cars, trucks, and trailers, along with stationary crates, cones, barrels, sticks, and other similar objects. In many cases, the pedestrians experience a period of partial to full occlusion by these objects or each other. Several variations of the

scenario also include one or two people walking in front of the vehicle, weaving between each other and occasionally going out of the field of view.

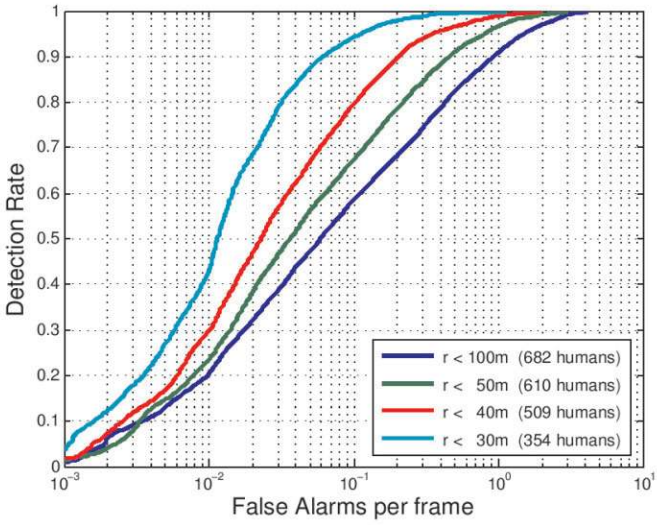
The imagery was manually ground-truthed by annotating a bounding box around each person in the left image of each frame, to a range of approximately 100 m. In total, our corpus includes approximately 6,000 annotated frames with approximately 10,000 annotated people, although we restrict our analysis to specific datasets which are representative of operational scenarios. Although people are annotated regardless of their posture or degree of occlusion, we only consider people who are in an upright posture with less than 50% occlusion for our analysis. We use the measure of the area of the intersection over the area of the union of the annotated and detected bounding boxes to declare a correct detection. However, for these datasets, we found that relaxing the common evaluation criteria of 50% intersection-over-union to 25% produced more meaningful results. This is because we are interested in detection at relatively long range where the segmentation error is dominated by the foreground fattening effect of stereo matching. As the scenes are relatively uncluttered, using a looser matching criteria still remains representative of actual detections. In order to present results that are meaningful when developing a complete, autonomous system capable of safe navigation, we present our results as the Pd, defined as the number of detections divided by the true number of people in the scene, versus the FAPF, defined as the number of incorrect detections divided by the number of frames in the dataset. We have observed that pixels-on-target are a dominating factor to classification performance, so we also illustrate the performance as a function of range, restricting the detections and annotations to several maximum ranges. This provides an indication of how the algorithm will perform with different resolution imagery or different camera or sensor configurations.

To demonstrate the effectiveness of our feature set and classifier, we first present results on a cross-validation test over many of our datasets. Figure 6(a) shows the performance of the system as an average of 1,000 trials on a dataset combined from many different scenarios, totaling 4,396 frames with 3,409 annotated people. From these sequences, 21,824 ROIs were extracted and each curve was generated by randomly selecting 80% of these ROIs for training and using the remaining 20% for testing. The resulting number of effective frames in each test sequence is thus 879, and the average number of humans is shown in the plot for the respective range restriction. For this test, no temporal filtering was used to adjust the classification scores. Figure 6(b) shows a sample of the images of the sequences used. The detections shown are indicative of the performance of the system (but are, in fact, based on a system trained without that sequence). Across our datasets, the system can achieve a 95% Pd at 0.1 FAPF for people less than 30 m and 85% Pd at 0.1 FAPF for people less than 40 m. For people out to 50 and 100 m, the system achieves 95% and 90% Pd respectively at 1 FAPF.

As the cross-validation results sample across all of the datasets being tested, they do not necessarily provide compelling evidence that the system is effective in new, unseen scenarios. To demonstrate that our system is robust in new environments, we show the performance on individual sequences that have never been used for training. Although less statistically significant, they are perhaps more indicative of the performance to be expected of the fielded system. Figure 7(a) and (b) show the results of the system without temporal filtering on two sequences held out from the training data (for which example images with detections are shown in Figure 7(c) and (d)). The same system was run on both datasets with no modification. As the plots show, the sequence shown in Figure 7(a) and (c) is more difficult than Figure 7(b) and (d), containing more clutter and occlusion. The system achieves well above 95% Pd at 0.1 FAPF for pedestrians less than 30 m and 80% Pd for less than 40 m. For a fielded system, we generally run at an operating point closer to 0.02 FAPF, which results in 90% Pd for $<$30 m and 65% Pd for $<$40 m, and maintain some degree of persistence of detected objects, propagating them with their predicted velocity for path planning.

The main source of false alarms of our system in these environments is due to the over segmentation of vehicles. An example of a false alarm on the front of a pickup truck is shown in the lower image of Figure 6(b). The individual distracter objects, such as barrels, tripods, and sign posts are only occasionally misclassified because they are normally segmented correctly. The main source of missed detections is due to variability of the stereo range data at long range, partial occlusion, and occasionally due to imprecise localization of the person due to under or over segmentation. Our system has some robustness to partial occlusion, but tends to break down after greater than 50% occlusion. The sequence shown in Figure 8 shows several examples of performance on occluding and overlapping people. The people in the near field are detected when they are unoccluded, or only slightly occluded. They are not detected when partially occluded either vertically (due to crossing the other person) or horizontally (due to the posts). Note, however, that the people are all tracked throughout the sequence (although with one incorrect association). The people in the far field are similarly not detected when they are partially occluded by the vehicles (or too far away), but are detected when they emerge into the open. The failure to detect partially occluded people is understandable because we only train a single classifier with data that does not contain many occluded people. An approach to addressing partial occlusion might be to train multiple classifiers for the different types of occlusion expected (lower torso, upper torso, left side, etc.). Alternatively, a parts-based classifier could be used to detect distinct portions of people. However, this approach would likely require a ground-plane assumption in order to detect the body parts correctly.

In addition to testing on ground-truthed datasets, the end-to-end system has been integrated into several systems for



(a)

(b)

(c)

(d)

Fig. 6. (a) The performance resulting from 1,000 trials of 80%/20% split cross-validation tests on 4,396 frames drawn from various scenarios. (b) Examples of images and detections from the various scenarios, with an example false alarm on the truck in the bottom image. The yellow boxes are detections, with a green overlay of the segmented person.

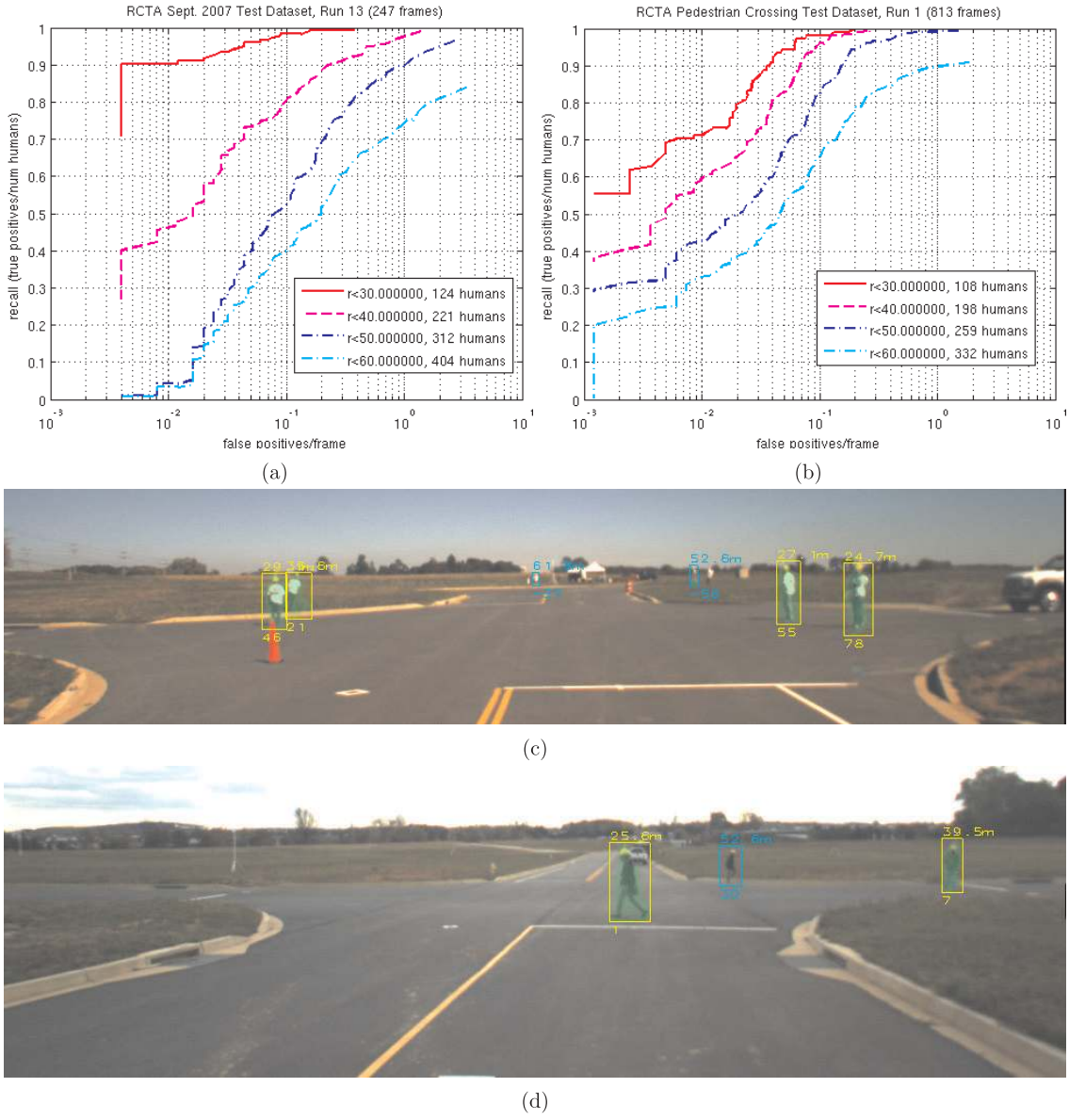


Fig. 7. (a) The performance for the run shown in (c), and (b) the performance for the run shown in (d). The yellow boxes are detections, with a green overlay of the segmented person; the cyan boxes are missed detections.

live testing. An earlier version of the system was fielded as part of the RCTA program SafeOps test, as reported by Bodt (2008). The system described here has been integrated onboard the test vehicle for an upcoming test, for which results will be published in the future. The system has also been used to demonstrate autonomous navigation in a lightly cluttered dynamic environment on a small vehicle (with cameras at approximately 1 m high and with a 12 cm baseline) traveling at approximately 1 m s^{-1}.

4.2. Urban Datasets

To illustrate that our system is competitive with other state-of-the-art stereo-based pedestrian detection systems, we also evaluated our system on datasets published by Ess et al. (2007, 2008). These datasets consist of 640 × 480 resolution color Bayer tiled imagery, taken at 15 Hz, with a 40 cm baseline camera pair pointed straight out at a height of approximately 1 m. The scenarios are significantly more complex than the semi-urban data, with many people in a busy shopping district in Zürich, Switzerland, with significant occlusion, clutter,

Fig. 8. A sequence of frames showing detections (yellow boxes, with green overlay the segmented person) and misses (cyan boxes) for people under occlusion. The number above the boxes indicates the range, and the number below indicates the track ID.

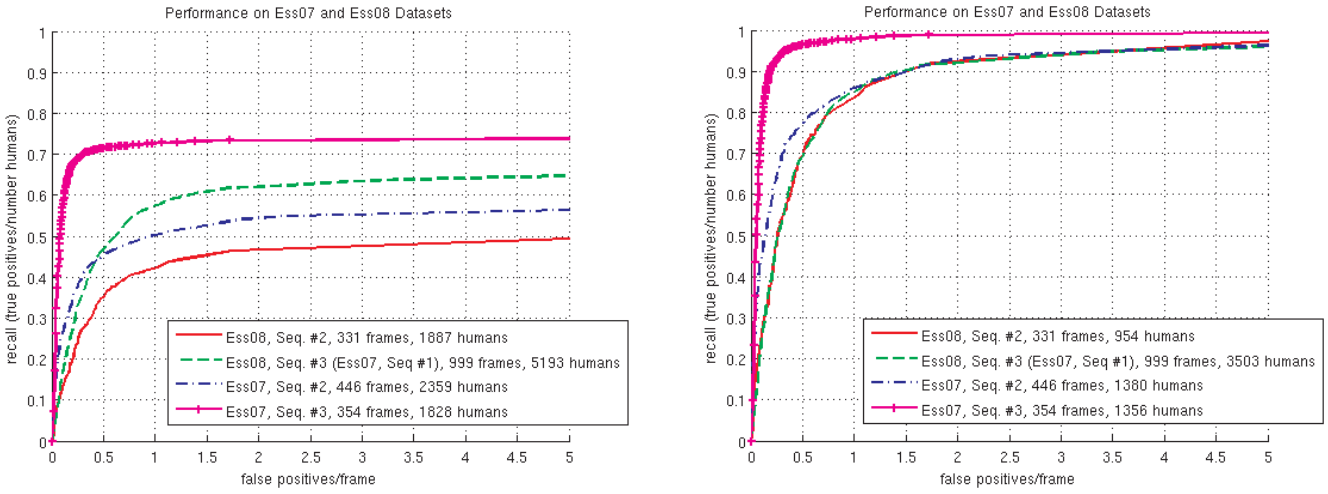


Fig. 9. (a) The performance curves for sequences from Ess et al. (2007, 2008) presented with the same evaluation criteria as their work. (b) The performance curves for the same sequences when all annotation that have less than 10% stereo coverage are eliminated, indicating that most of the misses in (a) are due to lack of stereo depth data on the people.

and motion. The annotations include all people whose torso is partially visible, and include children and partially upright postures, but not people sitting. To make a direct comparison to the results published by Ess et al., we use his detection criteria (50% intersection-over-union) and restrict the annotations used in the same way they do (with height greater than 80 pixels for sequence 2 of the 2008 data, and 60 pixels for all other data). We completely omit sequence 1 of the 2008 data because we were unable to generate acceptable stereo depth maps based on the camera models provided. The depth data density on all other sequences is acceptable, but not as dense as it could be, and results in reduced performance as discussed later. For direct comparison, we also train on exactly the same data as well (sequence 0 of the 2007 data).

The performance curves of our end-to-end system with the Ess et al. test sequences using exactly the same evaluation criteria are shown in Figure 9(a). Although the performance does not appear very good (between 0.4 and 0.7 recall at 1 false positive per frame, and with maximum achievable recalls between 0.5 and 0.75), it is very similar to the results reported by Ess et al.. In fact, the results are slightly better at 1 FAPF on all sequences except sequence 2 of the 2008 data (which is due to less stereo coverage). Examples of the scenes, along with stereo and the predicted velocity of certain pedestrians, are shown in Figures 10 and 11. Note that people are detected when they are in various poses or stages of walking and while carrying bags or briefcases. The main cause of the missed detections is simply due to a lack of stereo depth data density on people who are either too close or occluded. To illustrate this point, we also show the performance for the sequences where annotated people must have at least 10% stereo coverage (of the pixels defined by the annotated bounding box) in Figure 9(b). As our system relies on stereo data for both detection and classification, it can never find these people, nor would it be able to localize them to plan around them in a fully autonomous mode.

Our system misses detections and produces false positives in some understandable situations. For instance, it misses most children (left image of Figure 10), which were not included in any training data, and detects mannequins in shop windows or reflections of people in windows (right image of Figure 10). However, the majority of false detections is due to patchy stereo on flat surfaces such as buildings or cars, which results in the objects being over-segmented into a human-sized objects (as seen on the car in the left image of Figure 11). Many times, this results in false positives high up on buildings (as seen in the center image of Figure 11), that could be removed by only considering people who might enter the street or be a danger. In other cases, explicitly detecting other objects such as cars would remove the false detections. Despite not designing for many of these situations, our system is capable of achieving competitive performance while running at 10 Hz on the 640 × 480 imagery.

5. Conclusion

The results of our stereo-based pedestrian detection system show it to be effective at detecting people out to a range of 40 m in semi-urban environments. It achieves results comparable with alternative approaches with other sensors, but offers

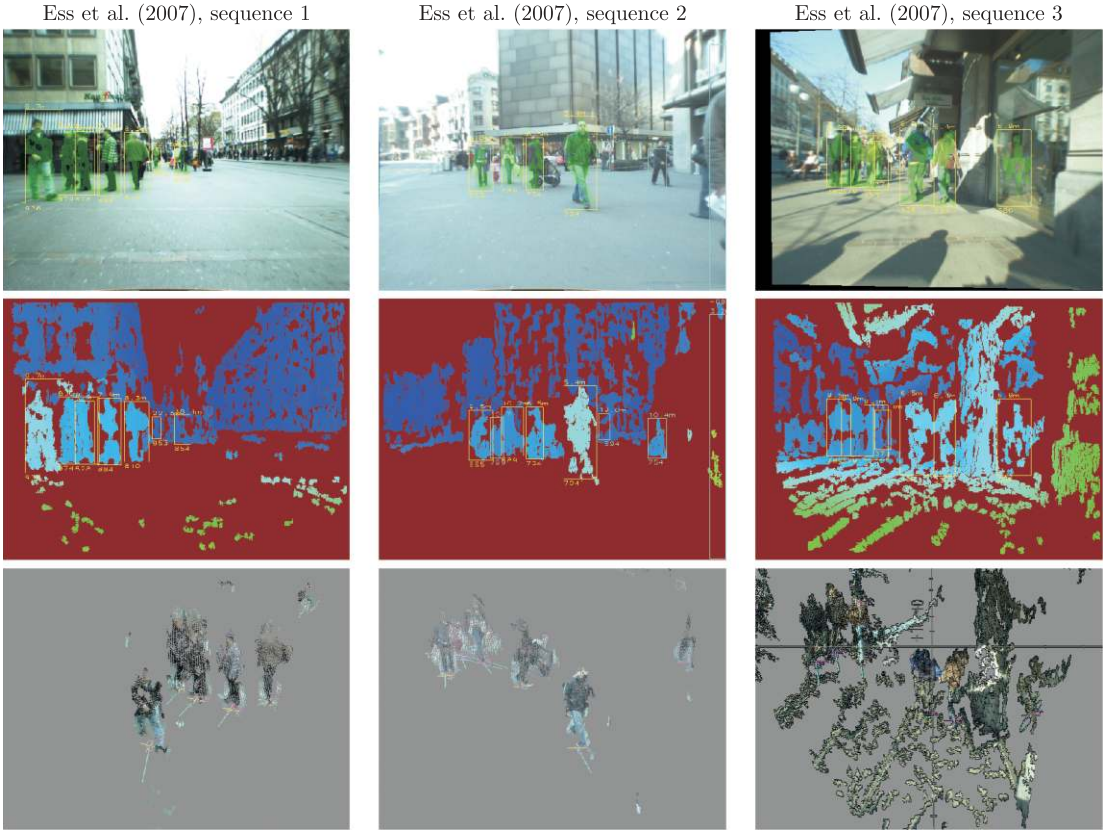


Fig. 10. Examples of detections (yellow boxes, with green overlay of segmented people) and misses (cyan boxes) (top row), the corresponding depth map (middle row) and velocity estimates on the three-dimensional point cloud (bottom row) for sequences from Ess et al. (2007). The false detection in the sequence 3 example is due to a reflection in the window.

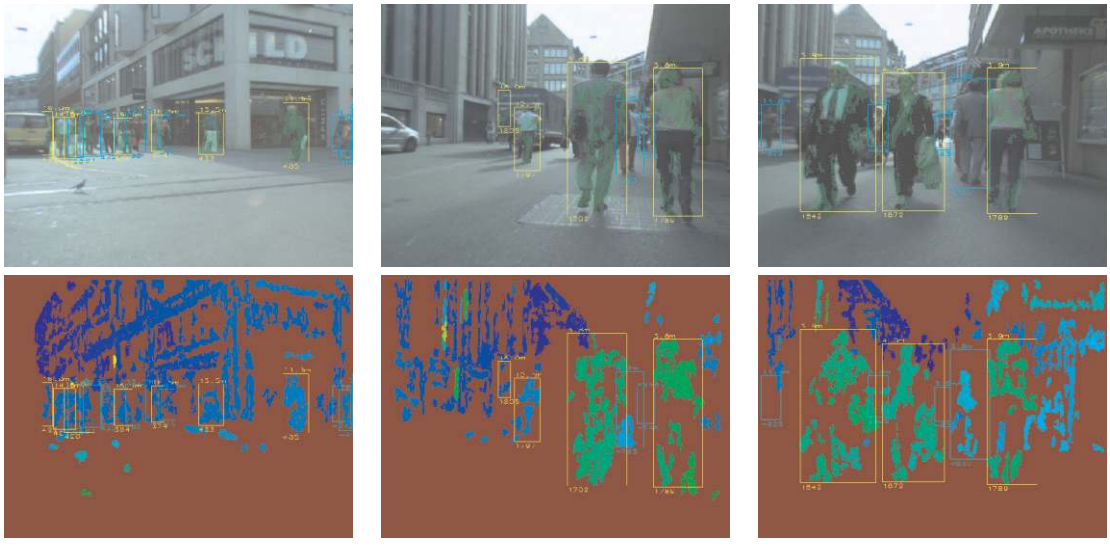


Fig. 11. Examples of detections (yellow boxes, with green overlay of segmented people) and misses (cyan boxes) (top row), the corresponding depth map (middle row) for sequence 2 from Ess et al. (2008). There are false alarms on the car in the left image and the bus in the middle image. The misses are generally due to lack of stereo coverage or excessive clutter.

the potential for long-term scalability to higher spatial resolution, smaller size, and lower cost than other sensors. It also performs similarly to state-of-the-art results from recent literature, while running significantly faster.

Our system can be improved in many ways, but we have identified several specific approaches that we feel would be most beneficial. In particular, adding appearance and motion features could substantially improve the detection rates. At close range, where there are many pixels-on-target, using appearance features will help to detect people under partial occlusion and in non-upright postures. At long range, using motion to segment moving objects from the background will help to increase the detection rate (although one cannot rely on motion exclusively, since stationary pedestrians are in as much danger as moving people). All of these techniques will benefit from increased camera resolution, but doing so will increase the computational cost. This strongly motivates the study of methods for efficiently focusing attention of specific areas of high resolution imagery.

Acknowledgments

The research described in this publication was carried out at the Jet Propulsion Laboratory, California Institute of Technology, with funding from the Army Research Lab (ARL) under the Robotics Collaborative Technology Alliance (RCTA) through an agreement with NASA. We would like to thank Piotr Dollar for providing us with his ground-truthing tool and David Trotz and Toni Ivanov for their help in ground-truthing our data.

References

Abd-Almageed, W., Hussein, M., Abdelkader, M. and Davis, L. (2007). Real-time human detection and tracking from mobile vehicles. *IEEE Intelligent Transportation Systems Conference*.

Arndt, R., Schweiger, R., Ritter, W., Paulus, D. and Lohlein, O. (2007). Detection and tracking of multiple pedestrians in automotive applications. *IEEE Intelligent Vehicles Symposium*.

Bajracharya, M., Moghaddam, B., Howard, A. and Matthies, L. (2008). Detecting personnel around UGVs using stereo vision. *Unmanned Systems Technology X* (*Proceedings of SPIE*, Vol. 6962). Bellingham WA , SPIE.

Bertozzi, M., Broggi, A., Rose, M. D., Felisa, M., Rakotomamonjy, A. and Suard, F. (2007). A pedestrian detector using histograms of oriented gradients and a support vector machine classifier. *IEEE Intelligent Transportation Systems Conference*.

Beymer, D. and Konolige, K. (1999). Real-time tracking of multiple people using continuous detection. *International Conference on Computer Vision (ICCV)*.

Bodt, B. A. (2008). Detecting and tracking moving humans from a moving vehicle. *Unmanned Systems Technology X* (*Proceedings of SPIE*, Vol. 6962). Bellingham WA , SPIE.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ess, A., Leibe, B. and Gool, L. V. (2007). Depth and appearance for mobile scene analysis. *International Conference on Computer Vision (ICCV)*.

Ess, A., Leibe, B., Schindler, K. and van Gool, L. (2008). A mobile vision system for robust multi-person tracking. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Eveland, C., Konolige, K. and Bolles, R. C. (1998). Background modeling for segmentation of video-rate stereo sequences. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 601–608.

Fuerstenberg, K. C., Dietmayer, K., and Willhoeft, V. (2002). Pedestrian recognition in urban traffic using a vehicle based multilayer laserscanner. In *IEEE Intelligent Vehicles Symposium*.

Gavrila, D. M. and Munder, S. (2007). Multi-cue pedestrian detection and tracking from a moving vehicle. *International Journal of Computer Vision*, 73(1): 41–59.

Goldberg, S., Maimone, M. and Matthies, L. (2002). Stereo vision and rover navigation software for planetary exploration. *IEEE Aerospace Conference*.

Howard, A. (2008). Real-time stereo visual odometry for autonomous ground vehicles. *IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*.

Howard, A., Matthies, L., Huertas, A., Bajracharya, M. and Rankin, A. (2007). Detecting pedestrians with stereo vision: safe operation of autonomous ground vehicles in dynamic environments. *International Symposium of Robotics Research*.

Liebe, B., Cornelis, N., Cornelis, K. and Gool, L. V. (2007). Dynamic 3D scene analysis from a moving vehicle. *IEEE Conference Computer Vision and Pattern Recognition (CVPR)*.

Ma, G., Park, S., Ioffe, A., Muller-Schneiders, S. and Kummert, A. (2007). A real time object detection approach applied to reliable pedestrian detection. *IEEE Intelligent Vehicles Symposium*.

Navarro-Serment, L. E., Mertz, C. and Hebert, M. (2008). LADAR-based pedestrian detection and tracking. *Workshop on Human Detection from Mobile Platforms, IEEE International Conference on Robotics and Automation (ICRA)*.

Sabzmeydani, P. and Mori, G. (2007). Detecting pedestrians by learning shapelet features. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Seeman, E., Fritz, M. and Schiele, B. (2007). Towards robust pedestrian detection in crowded image sequences. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Shashua, A., Gdalyahu, Y. and Hayun, G. (2004). Pedestrian detection for driving assistance systems: single frame classification and system level performance. *IEEE Intelligent Vehicles Symposium*.

Sotelo, M., Parra, I., Fernandez, D. and Naranjo, E. (2006). Pedestrian detection using svm and multi-feature combination. *IEEE Intelligent Transportation Systems Conference*.

Thornton, S. M., Hoffelder, M. and Morris, D. D. (2008). Multi-sensor detection and tracking of humans for safe operations with unmanned ground vehicles. *Workshop on Human Detection from Mobile Platforms, IEEE International Conference on Robotics and Automation (ICRA)*.

Tomiuc, C., Nedevschi, S. and Meinecke, M. M. (2007). Pedestrian detection and classification based on 2d and 3d information for driving assistance systems. *IEEE Intelligent Computer Communication and Processing Conference*.

Tuzel, O., Porikli, F. and Meer, P. (2007). Human detection via classification on Riemannian manifolds. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Viola, P., Jones, M. J. and Snow, D. (2003). Detecting pedestrians using patterns of motion and appearance. *IEEE International Conference on Compuer Vision*.

Wu, B. and Nevatia, R. (2007). Simultaneous object detection and segmentation by boosting local shape feature based classifier. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhao, T., Nevatia, R. and Wu, B. (2008). Segmentation and tracking of multiple humans in crowded environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(7): 1198–1211.