Precise Tracking using High Resolution Real-time Stereo

M. Usman Butt
Electrical & Computer Engineering,
The University of Auckland, New Zealand
Email: ubut002@aucklanduni.ac.nz

John Morris
Electrical & Computer Engineering,
The University of Auckland, New Zealand
Email: j.morris@auckland.ac.nz

***Abstract*—We describe a method to detect and precisely track an object using high resolution 3D data obtained from a real-time stereo system. The precision of the data was expected to make tracking objects through mergers and occlusions easier as close objects can still be separated over large areas of the scene, however it introduced some unforeseen problems in tracking. In particular, choosing a suitable single reference point to represent the position of the person tracked was made harder because a person's 3D structure was usually manifest. Experiments indicated that a composite reference point consisting of centroid X- and head Y- and Z- coordinates was the best choice. A Kalman filter was able to accurately predict the object position in the next frame. It was also able to flag a cut off head (a common problem in our high resolution stereo due to low contrast in the shadows of the neck) and ensure that it was re-attached to its body. Results show that our system was able to track precisely at 20 m distance with tracking error $\lesssim$ 50 mm.**

I. Introduction

Detection and tracking of rigid objects (*e.g.* vehicles) and non-rigid objects (*e.g.* people) in dynamic environments has many applications including human activity monitoring, traffic monitoring, surveillance and security and service robots. However, abrupt changes in object motion, changing illumination, changing object appearance, self occlusion (entering shadows), object-to-object occlusion and camera motion make real-time object detection and tracking a challenging task. A tracking system ideally determines the trajectory of each object in the scene for further analysis. Two kinds of sensors are used: (i) *active sensors* transmit position to a receiver using markers fixed on the tracked object and (ii) *passive sensors e.g.* video cameras. Active sensors are more reliable, but markers may be fixed on targets only in very limited applications. Thus, vision based detection and tracking is very important to many applications. Video cameras are used to locate and track objects. Although, there exist techniques to locate people in 3D world with a single camera [1], yet they are not reliable and precise. Stereo cameras not only provide the real-world position but are also good for segmentation of objects in the presence of occlusion and shadows. Many techniques have been developed to locate and track objects using low resolution stereo images. However, low resolution images provide less information about the environment and 3D information obtained from theses images is less precise. So, it is hard to separate occluding objects in the distance with low resolution which results in merged objects. In this paper, we use high resolution depth map obtained from a high resolution real-time stereo system to locate and track an object more precisely. Before the object is tracked, object segmentation and determination from the scene is a major challenge. Other stereo tracking techniques rely on other features such as motion, intensity or color along with depth feature for object segmentation. We fully exploit and rely on only 3D data for object segmentation and tracking. We use dense depth (disparity) map obtained from Symmetric Dynamic Programming Stereo (SDPS) implemented on real-time FPGA stereo system. We exploit interesting triangular profile of SDPS depth map which gives best outline (contour) of object of interest from the scene. Another challenge in object tracking is the selection of an adequate reference point which corresponds to the object location in 3D-world and is smooth and stable for tracking articulated objects. We propose a simple approach for object location and tracking in the real-world with a reference point comprising Y and Z coordinates from the head of an articulated object and X from its centroid. An approach which simply down-samples and produces object outline might have been good but throws away data. It can be used in separation of merged objects. So, it was necessary to keep all the data.

A. Related Work

The large number of applications for vision based tracking has led to an equally large background literature: Yilmaz *et al.* survey trends in object tracking algorithms until 2006[2]. Here, we only mention 3D vision based tracking algorithms. The first step in passive sensor based tracking - segmentation of relevant objects from the scene - is considered the most difficult and solutions depend on the application. When the cameras are mounted on a moving platform then structure from motion is the most common approach for background modeling and automatic camera calibration [3], [4]. Various additional cues are also necessary to locate an object of interest, including disparity maps, ground plane estimation, vertical edge and disparity symmetries [5], optical flow [6], [7] *etc.*Gandhi and Trivedi discuss the combination of these cues and other types of sensor which could be used to track objects like pedestrians [8].

For cameras mounted on a fixed platforms, the background

is mostly static - with relatively small numbers of dynamic objects, for example trees - and the most common approach is simple background subtraction. Variation in pixel intensity from changing ambient light makes this difficult. Therefore subtraction of intensities in consecutive frames is rarely satisfactory. Wren *et al.* independently model each pixel over time through Gaussian probability density function [9], [1]. A better approach would be to apply a temporal median filter, but this demands large memory to store history from previous frames [10]. A more memory efficient method keeps a histogram for each pixel over time [11]. Piccardi has reviewed the accuracy and performance of background subtraction methods [12].

The next challenge is to identify the region in which the object of interest lies. There are three general strategies - point tracking [13], kernel tracking [14] and silhouette tracking [15]. Dense optical flow is another solution - points with similar motion vectors should correspond to the same object [11]. However, this is unlikely to hold for non-rigid objects like pedestrians merging or occluding each other. 3D data from stereo images has the potential to overcome many tracking problems: the main drawback is computation demand. Recently, Cai *et al.* described a sparse feature based stereo tracking system[16] . They obtain a sparse depth map of matched feature points, detected by a Harris operator, from stereo images and project it on to a ground plane. Kernel-based clustering was used to identify relevant objects and track them. In contrast, our system, in which attached hardware produces dense disparity maps, does not need to constrain itself to a small number of features and can use full 3D maps of each person to recognize individuals and separate them in clusters.

II. Symmetric Dynamic Programming Stereo (SDPS)

Our FPGA system uses Gimel'farb's SDPS algorithm [17] and outputs disparity maps with a disparity range of 128 giving $\sim 1\%$ depth accuracy at 30 fps [18], [19] .

SDPS matches scanline by scanline and produces the disparity map seen by a Cyclopæan eye from a point half-way between the left and right cameras [17]. In this view, transitions between disparity levels must pass through monocularly visible states, *i.e.* any change in disparity along a scanline will be accompanied by one monocular point for each unit change in disparity: if the disparity changes from d to $d+a$ along a scanline, ML or MR pixels with disparities, $d{+}1, d{+}2, ..., d{+}a$ will always appear. This 'triangular' profile does not always match the true profile, but it is always a valid candidate because pixels on the slope $(d \rightarrow d+a)$ are monocular: we have no depth information for them and *the triangular profile is as good a hypothesis for the true profile as any*. Thus objects separated from their backgrounds appear with 'outlines' (bands of ML or MR points) in the occlusion map. Objects well separated from the background or other objects have broader outlines.

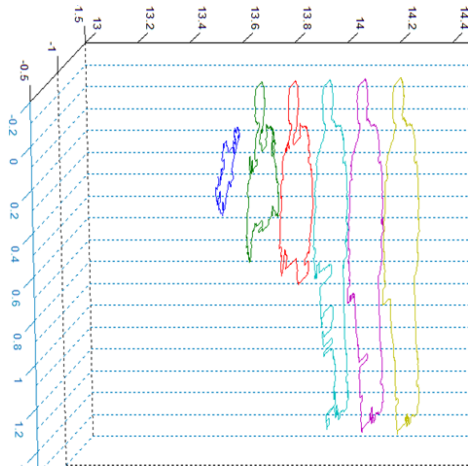

Figure 1. Human contours, converted into Real-world coordinates, at neighbouring disparities

III. Object Detection and Tracking

Generally, a camera viewing a crowd scene (*e.g.* to gather information on traffic patterns) or a secure area will be placed above the region monitored. Thus the flat ground plane will appear in disparity maps as bands of equal disparity areas. We remove it by setting the disparity in the ground region (where the scene and reference images show the same disparity) to 0 - effectively moving the ground plane to infinity.

To filter out irrelevant objects (*e.g.* dogs, cats, *etc.*) and make the problem computationally tractable, we assume that 'objects of interest' can be any rigid (e.g. vehicles) or non-rigid (*e.g.* human) object, but object sizes must lie in a pre-defined range. For example, for humans, the silhouette (height×width) of a standing person (arms by side) ranges from $0.8 \times 0.27 = 0.22 \;\; m^2$ (a one year old walking) to $2.47 \times 0.8 = 1.97 \;\; m^2$ (the tallest recorded man[20]). However, objects may deform (*e.g.* a human walking) but the silhouette will remain within some range. We assume that object speed is less than $10 \;\; m \;\; s^{-1}$ (Olympic sprinter), the ground plane is basically flat and fixed and cameras are tilted at fixed angle to the ground plane. A tracked object is isolated and moving upright.

A. Object Detection

An object's 3D structure is represented by several contours at neighbouring disparities in our high resolution depth maps (see Figure 1).

A key step is the determination of the contour best outlining the object. We choose the largest area contour containing a significant number of binocularly visible points: when the ratio of the areas of two adjacent contours, $A_{final} = A_d/A_{d-1}$ (where A_d and A_{d-1} are the areas of contour at disparities d and $d{-}1$, respectively) exceeds some threshold (currently 0.9), we consider it the 'true' outline of the object: it represents the 'face' of an object seen by both cameras. Thus contours representing the separation between an object and its background,

which closely follow the true object outline and are thus only slightly larger in area, are ignored.

The expected range of values for a human silhouette (area, $A_{ref}(d)$; width, $W_{ref}(d)$; and height, $H_{ref}(d)$) projected onto the image plane is pre-computed for each disparity, $d = p_{min}, \ldots, p_{max}$ and stored in a table. For the current hardware, $p_{min} \geq 0$ and $p_{max} \leq 127$.

The background is removed from the image:

$$p(x, y) = 0 \quad \text{if} \quad p(x, y) <= p_{ref}(x, y) \tag{1}$$

where $p(x, y)$ and $p_{ref}(x, y)$ are pixels in the image and reference disparity maps, respectively.

From a histogram of the disparity map, we select the front-most disparity level, d_s, which could contain a relevant object. Then the disparity map is thresholded from foreground disparity value, $d_s = p_{max}$, to the background disparity value, d_{back} = p_{min}.

At each disparity level, the noise from background subtraction and SDPS generated streaks is reduced using Morphological Operation (MO) and contours $B_{d,j}$, where d is the disparity and j is a contour index, are computed using a border following algorithm [21]. Other approaches apply MO to the original disparity map[22], [23], [24], potentially distorting it and the real world coordinates computed from it. We apply MO only to one disparity level at a time because it has the capability to remove a known and understood defect of SDPS - the streaks. The area, $A_{d,j}$, centroid, (c_x, c_y) and bounding box, (x, y, w, h) of each contour are calculated.

Contours which could represent objects (*i.e.* which are larger than pre-computed silhouette sizes) are labelled *candidate contours* for isolated (single) objects.

Binocularly visible 'true' contours are determined by comparing the candidate contours at adjacent disparity levels. Contours may represent new (previously undetected background) objects or part of an existing object if they are inside an existing object.

The rules are:

1) If the centroid of $B_{d,j}$, lies inside $B_{d-1,k}$, then, $B_{d,j}$ is enclosed by $B_{d-1,k}$, then
2) If the heights of both the contours are same within the tolerance, $(h_{d,j} - h_{d-1,k} < \varepsilon_h)$ and the internal contour area ratio, $\frac{A_{d,j}}{A_{d-1,k}} > A_{final}$ (currently $A_{final} = 0.9$), then $B_{d-1,k}$ is an *object outline*

Enclosed contours are retained as attributes of an object as they provide information about the 3D structure of the object. Objects identified from the disparity map are reprojected onto real-world coordinates using calibration parameters. The real-world object outlines provide the object dimensions (width and height) in addition to the object location. For object location we select a reference point (described in the next section) as the tracked point representing an object. As each new object is detected, it is assigned a label and its velocity is set to zero. A final list of objects along with their attributes is returned at each frame and used in conjunction with the list of objects in next frame for tracking.

B. Tracking

Objects detected in each frame are tracked from frame to frame using real-world size and location information. A Kalman filter is used to predict the object location in the next frame[25]. We assumed that objects move with constant velocity. The filter's state vector is $[X, Y, Z, dX/dt, dY/dt, dZ/dt]$. A major drawback of SDPS is the generation of streaks from bad matching in low texture regions. This causes heads to be cut off, especially at low depth resolution. The predicted location not only helps to interpolate when objects are missed (due to occlusions or mergers) but also identifies heads cut off by SDPS streaks. We compare the predicted peak 'Y' coordinate with the observed one to determine whether the head is missing. If it is, then we retrieve the head contour by searching over a small region above the object torso and joining it to the remaining body contour. Further, by translating the object outline at predicted position in the real-world, a predicted disparity map by perspective projection of translated object outline is generated. The object in the current frame is matched if it has maximum overlap with the predicted shape at that location.

Curiously, our high resolution 3D data makes choosing a suitable *single* point to represent the tracked object harder. A centroid is an obvious choice, but for articulated objects, a centroid moves as the object deforms. Other choices are the feet, *i.e.* the ground point of the object outline, or the head. As people walk, their point of contact with the ground changes significantly. Moreover, at least one foot usually touches the ground, so a portion of the feet are cut by background subtraction. Thus, selecting a suitable point in the feet region is difficult.

To select an adequate reference point, we tracked a single isolated object in a video sequence (for sequence characteristics see Section IV) with three different reference points: head, feet and centroid. We checked the Kalman filter predicted positions with the observed positions (see Figure 2). Figures 2a), b), and c) present observed and predicted (filtered) tracks along X, Y and Z for each reference point. Deviation of measured coordinates from the predicted values are approximately normally distributed and presented in Figure 2d), e) and f). Comparing the coordinates of each reference point, first we see that the X coordinate of centroid (Figure 2a) has small variations and the least $\sigma_{Xc} = 0.04$ compared to the head ($\sigma_{Xh} = 0.05$) or feet ($\sigma_{Xf} = 0.08$). Whereas the head Y coordinate has the smallest variation and $\sigma_{Yh} = 0.04$ compared to $\sigma_{Yc} = 0.09$ for centroid and $\sigma_{Yf} = 0.07$ for feet points. The Y coordinate also helps us to identify a missing head in the next frame. The head Z coordinate has the least deviation ($\sigma_{Zh} = 0.14$). So, to form a trackable reference point, we select the head Y and Z coordinates and centroid X, which actually represents a point on the head directly above the body centre.

Isolated objects are matched frame to frame by checking that their attributes (area, height and width) match within some tolerance and the object has travelled a physically possible distance between frames.

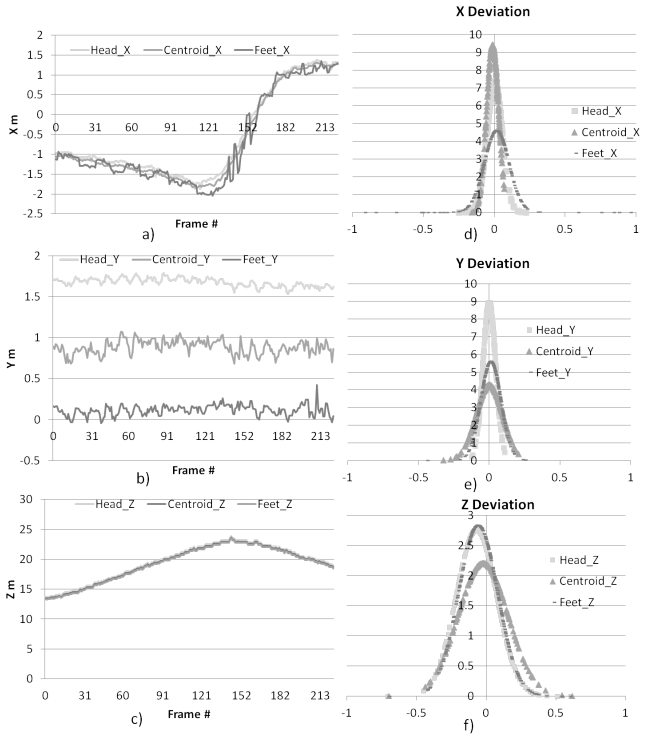

Figure 2. Comparison between a) X-, b) Y-, c) Z-coordinates of head, feet and centroid reference points for tracking with deviations from the predicted points along d) X, e) Y and f) Z.

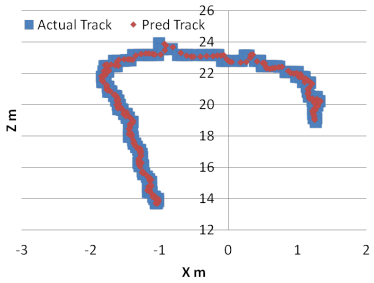

Figure 3. Bird's eye view of observed and predicted tracks: Scenario A single person

IV. Results

We assessed tracking precision on three different sequences described below.

For first two scenarios (A and B), two identical cameras with 16 mm lenses on a 440 mm baseline were mounted at a height of 5.2 m pointing down 15° to the ground. They cover a viewing area 2 m wide at 10 m from the cameras to 8 m at 25 m depth. For the third scenario, cameras with 9 mm lenses on a 200 mm baseline were mounted at a height of 1.65 m and an angle of 9° down covering a smaller viewing area. The system was calibrated using Bouguet's procedure[26].

Scenario A: In this scenario, a single isolated subject 'A' starts at (-1, -, 13.7), moves to (1.7, -, 22.4), turns around and walks via (1.1, -, 21.6) to (1.2, -, 18.9) on the X-Z ground plane

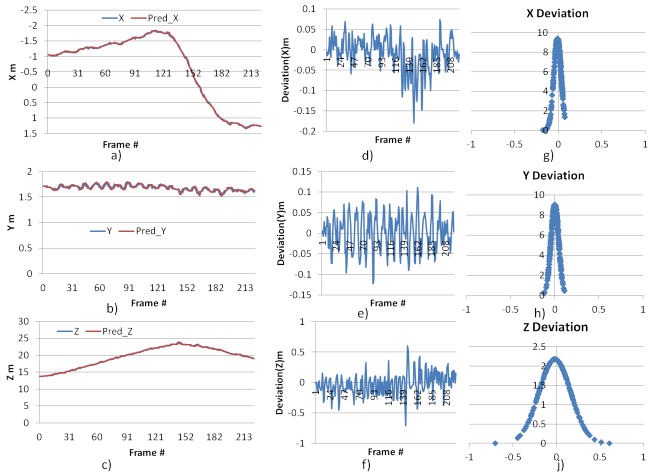

Figure 4. Scenario A - Observed and predicted tracks a)- c), error of observed data from the predicted d)-f) and normal distribution of error g)-j) along X, Y and Z coords.

i.e. as shown in Figure 3. The object is standing vertically, so we do not specify the Y coordinate. Individual X, Y and Z coordinates of the tracked subject reference point are presented in Figure 4.

The graph shows observed (measured) position and Kalman filter predicted positions in each frame. The subject initially moves to the left (decreasing X) and then back across the system axis to the right (positive X). Deviation between predicted and observed values increases with increasing Z - a consequence of lower depth resolution at a distance. The natural rise and fall of a walking subject is clearly evident in the Y component's oscillatory behaviour. The Z coordinate shows the actual and predicted distance on the ground from cameras. Variations of observed positions from Kalman filter predicted positions along X, Y and Z coordinates are presented in Figure 4d), e) and f). As expected, the variation increases as the object moves away. For X- and Y- components this is due to the projective transform - one pixel translates to a larger distance as Z increases. Z-component variation increases because the reciprocal relation between disparities and distance implies lower Z resolution at a distance. Variations are approximately normally distributed with $\sigma_X = 0.04$, $\sigma_Y = 0.04$ and $\sigma_Z = 0.18$ - see Figure 4. Since we track a reference point at head of the body, the Y component represents the subject height. This subject's actual height was 1.7 m and measured height was 1.64 m ($\sigma = 50$ mm). Streaking artefacts and the natural gait contribute to this error (long 'flat' strides lower the head slightly), but it still represents precise tracking for a deformable object.

Scenario B: In this scenario, we tracked two isolated subjects (see Figure 5). Individual X, Y and Z coordinates for observed and predicted tracks of both subjects are presented in Figure 6. The heights obtained from the Y coordinate in Figure 6b) are A :1.72 m ($\sigma = 40$ mm) and B: 1. 67 m ($\sigma = 40$ mm) compared to actual heights of 1.73 m and 1.70 m. Subject A's

Original left image

Disparity map

Figure 5. Scenario B - Original scene

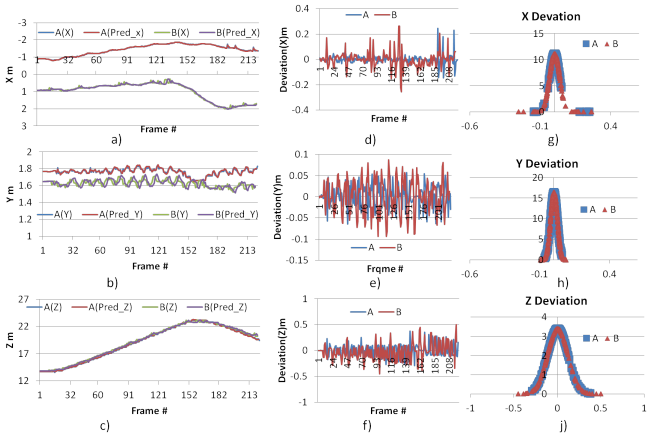

Figure 6. Scenario B - Observed and predicted tracks a)- c); deviations of observations from predictions d)-f) and distribution of deviations g)-j) for X-, Y- and Z-coordinates.

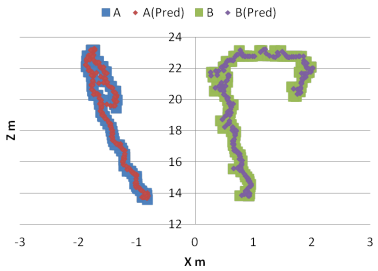

Figure 7. Bird's eye view of observed and predicted tracks: Scenario B - two people

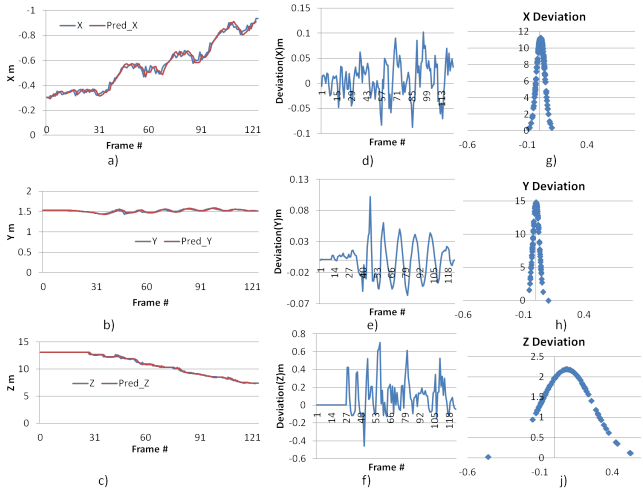

Figure 8. Scenario C - Observed and predicted tracks a)- c), error of observed data from the predicted d)-f) and normal distribution of error g)-j) along X-, Y- and Z- coordinates

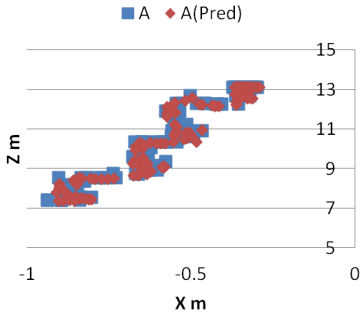

Figure 9. Bird's eye view of observed and predicted tracks of a single person in Scenario C

observed mean heignt is very close to the actual because he is walking with small steps and rising slightly onto his toes as he does. In contrast, subject'B's observed mean height is less than the actual because he is walking with longer strides on the soles of his feet. Observed and predicted tracks in the X-Z plane for both subjects are shown in Figure 7. 'A' starts at (-0.9, -, 13.7), moves to (1.8, -, 22.8), turns back and walks to (-1.3, -, 19.4). 'B' starts at (0.9, -, 13.8), moves to (0.4, -, 21.8), turns around and walks via (1.8, -, 22.7) to (1.7, -, 20.2).

Scenario C In this scenario, the subject starts about 13 m away and walks about 6 m towards the cameras. Our subject's arms are swinging very quickly resulting in significant shape variation. The subject was still successfully tracked over a sequence of 125 frames - see Figure 8. The Y coordinate of the reference point gives a mean height of 1.52 m ($\sigma = 30$ mm) compared to an actual height of 1.56 m. The swinging arms caused the X-component of reference point to move backwards and forwards in line with the shift of the centroid of the subject's silhouette projected onto the image planes. Our system has sufficient resolution to measure the 3D position of the arms at this distance, so it would be possible to move to a multiple limb articulated model of our subject if an application required it. The subject's track is presented in Figure 9.

V. Conclusion

We have demonstrated that a high resolution stereo system can track isolated objects effectively and precisely. The ability to generate a closely spaced contours from the high resolution images and depth map means that the problem of separating occluding or merged objects becomes easier - a lower resolution system will only see merged objects - but presented some challenges in finding a single reliable reference point as a basis for a simple tracking model. We were able to track with a standard deviation of a few tens of mm - enough to distinguish closely interacting people at 20 m distances. For tracking people in outdoor scenes with a slightly elevated camera system, a composite reference point consisting of centroid for X and head for Y and Z proved effective.

Precision tracking also provides the potential to track faster moving (*e.g.* bicyclists, traffic, ...) or smaller objects (*e.g.* items on fast conveyor belts).

We were able to correct for one artefact introduced by the real-time stereo - the guillotined head when it appeared in subsequent frames. However, if the object is first detected headless, the Kalman filter is initialized incorrectly and the system needs time to find the head. Since the head is invariably detected 'floating' above the torso, correcting this problem should be straightforward.

The next stage in this work is dealing with scenes containing more complex occlusions and merges using the precise tracking that we report here.

References

[1] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, p. 781, 1997.

[2] M. S. Alper Yilmaz, Omer Javed, "Object tracking: A survey," *ACM Computing Surveys*, vol. 38, No. 4, 2006.

[3] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 44–57.

[4] R. Mandelbaum, G. Salgian, and H. Sawhney, "Correlation-based estimation of ego-motion and structure from motion and stereo," in *iccv*. Published by the IEEE Computer Society, 1999, p. 544.

[5] U. Franke and A. Joos, "Real-time stereo vision for urban traffic scene understanding," in *Proceedings of the IEEE Intelligent Vehicles Symposium, 2000. IV 2000*, 2000, pp. 273–278.

[6] T. Dang, C. Hoffmann, and C. Stiller, "Fusing optical flow and stereo disparity for object tracking," in *Proc. IEEE Int. Conf. Intelligent Transportation Systems*, 2002, pp. 112–117.

[7] U. Franke, C. Rabe, H. Badino, and S. Gehrig, "6d-vision: Fusion of stereo and motion for robust environment perception," *Pattern Recognition*, pp. 216–223, 2005.

[8] T. Gandhi and M. Trivedi, "Pedestrian collision avoidance systems: A survey of computer vision based recent studies," in *IEEE Intelligent Transportation Systems Conference, 2006. ITSC'06*, 2006, pp. 976–981.

[9] T. Zhao and R. Nevatia, "Tracking multiple humans in complex situations," *Trans PAMI*, vol. 26, no. 9, pp. 1208–1221, September 2004.

[10] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1337–1342, 2003.

[11] O. Sidla, Y. Lypetskyy, N. Brandle, and S. Seer, "Pedestrian detection and tracking for counting applications in crowded situations," in *IEEE International Conference on Video and Signal Based Surveillance, 2006. AVSS'06*, 2006, pp. 70–70.

[12] M. Piccardi, "Background subtraction techniques: a review," in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, 2004, pp. 3099–3104.

[13] O. Javed and M. Shah, "Tracking and object classification for automated surveillance," *Computer VisionECCV 2002*, pp. 439–443, 2006.

[14] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 564–575, 2003.

[15] K. Cheung, S. Baker, and T. Kanade, "Shape-from-silhouette across time part ii: Applications to human modeling and markerless motion tracking," *International Journal of Computer Vision*, vol. 63, no. 3, pp. 225–245, 2005.

[16] L. Cai, L. He, Y. Xu, Y. Zhao, and X. Yang, "Multi-object detection and tracking by stereo vision," *Pattern Recognition*, vol. 43, no. 12, pp. 4028–4041, 2010. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2010.06.012

[17] G. L. Gimel'farb, "Probabilistic regularisation and symmetry in binocular dynamic programming stereo," *Pattern Recognition Letters*, vol. 23, no. 4, pp. 431–442, 2002.

[18] J. Morris, K. Jawed, and G. Gimel'farb, "Intelligent vision: A first step - real time stereovision," in *Advanced Concepts for Intelligent Vision Systems (ACIVS'2009)*, ser. LNCS, J Blanc-Tallon *et al.*, Ed., vol. 5807. Springer, 2009, pp. 355–366.

[19] J. Morris, K. Jawed, G. Gimel'farb, and T. Khan, "Breaking the 'ton': Achieving 1% depth accuracy from stereo in real time," in *Image and Vision Computing NZ*, D. Bailey, Ed. IEEE CS Press, 2009.

[20] Guinness World Records, *Guinness World Records*. Guinness World Records Ltd, 2010. [Online]. Available: {http://www.guinnessworldrecords.com}

[21] K. A. S. Suzuki, "Topological structure analysis of digitized binary images by border following," *Computer Vision, Graphics, and Image Processing*, vol. 30, pp. 32–46, 1985.

[22] D. Beymer and K. Konolige, "Real-time tracking of multiple people using stereo," in *Frame-Rate99*, 1999.

[23] T. D. T, G. G. G, M. Harville, and J. Woodfill, "Integrated person tracking using stereo, colour, and pattern detection," *Int J Computer Vision*, vol. 37(2), pp. 175–185, 2000.

[24] Y.-Y. L. Mau-Tsuen Yang, Shih-Chun Wang, "A multimodal fusion system for people detection and tracking," *Int J Imaging Syst Technol*, vol. 15, pp. 131–142, 2005.

[25] R. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[26] J.-Y. Bouguet, "Camera calibration toolbox for Matlab," www.vision.caltech.edu/bouguetj/calib$_$doc, 1999.