HEAD DETECTION IN STEREO DATA FOR PEOPLE COUNTING AND SEGMENTATION

Tim van Oosterhout, Sander Bakkes and Ben Kröse
CREATE-IT Applied Research, Amsterdam University of Applied Sciences (HvA)
Duivendrechtsekade 36-38, 1096 AH Amsterdam, The Netherlands
{*t.j.m.van.oosterhout, s.c.j.bakkes, b.j.a.krose*}*@hva.nl*

Keywords: Head detection, People counting, People tracking, Stereo cameras.

Abstract: In this paper we propose a head detection method using range data from a stereo camera. The method is based on a technique that has been introduced in the domain of voxel data. For application in stereo cameras, the technique is extended (1) to be applicable to stereo data, and (2) to be robust with regard to noise and variation in environmental settings. The method consists of foreground selection, head detection, and blob separation, and, to improve results in case of misdetections, incorporates a means for people tracking. It is tested in experiments with actual stereo data, gathered from three distinct real-life scenarios. Experimental results show that the proposed method performs well in terms of both precision and recall. In addition, the method was shown to perform well in highly crowded situations. From our results, we may conclude that the proposed method provides a strong basis for head detection in applications that utilise stereo cameras.

1 INTRODUCTION

Counting people from video streams is important in many applications such as bottleneck detection, waiting line measurement, interesting route detection and measurement of social behaviour. Numerous methods discussed in the literature are based on the assumption that people can be segmented from the background by means of appearance (Stauffer and Grimson, 1999) or motion properties (Heisele and Woehler, 1998). When people overlap in the image domain these blob based segmentation methods do not give an accurate count of the amount of people visible. To reduce overlap, cameras are mounted on the ceiling and are directed straight or slanted down. Methods have been proposed that use appearance cues in the 2D image for head detection (Fu et al., 2007; Ishii et al., 2004; Zhao and Nevatia, 2003). However, appearance methods are very sensitive to illumination conditions and colour similarity between subject and background.

Systems with stereo cameras have been proposed to solve a number of these shortcomings. A problem with object segmentation from stereo data is that stereo range data is generally noisy and as a result leads to incorrect segmentations. Moreover, even in clean stereo data many stereo correspondence algorithms produce artefacts known as foreground fattening (Scharstein and Szeliski, 2002) which may prevent nearby but separate objects from being correctly segmented. In this paper, we propose a tailored method to robustly detect and count people using data from stereo cameras. We extract features from the noisy range data in the form of sphere-shaped objects. We show that the method accurately performs head detection in applications with stereo cameras.

2 RELATED WORK

An approach to people counting in stereo is projecting all observations to the ground plane resulting in an occupancy map. (Beymer, 2000) uses a volume of interest located at head height. Points in this volume are projected and binned to obtain the positions of people. (Hayashi et al., 2004) use a full occupancy map and correct for the fact that people that are further away from the camera have less points by giving far points more influence on the occupancy map. Other segmenting methods in stereo include 3D clustering and region growing (Kelly et al., 2009) and connected component labelling based on depth layers combined with skin hue detection (Darrell et al., 2000).

One common technique in head detection is to look for an omega shaped contour in a 2D side-view. (Zhao and Nevatia, 2003) use edge detection to obtain the contours which are then matched against the

omega shaped template. (Park and Aggarwal, 2000) localise people in stereo, but use a partial ellipse fitting technique to find heads. (Luo and Guo, 2001) use stereo vision to segment the image into different depth layers after which they fit a contour.

A number of approaches are based on finding elliptical shapes in 2.5D (range) or 3D (voxel) representations. (Huang et al., 2004) uses scale-adaptive filtering in the range domain to find elliptical objects of predefined size. (Hoshino and Izumi, 2006) detect circles in one image from a stereo pair and project them onto the other to test hypotheses for their positions and radii. 3D data from multiple cameras can be used for body part localization (Mikić et al., 2003). In our system we use a shape based approach using stereo (range) data instead of a 3D voxel representation.

3 METHOD

Data from the stereo camera consists of a colour map and a depth map. The dynamic portion of both maps is determined using an adaptive background model. An algorithm is used that matches a spherical crust template on the foreground regions of the depth map. False positives are suppressed by putting constraints on the spread of the points within the template. Then blob separation is performed. In the last step the detections are fed into a tracker that ensures the continuity of individual detections.

3.1 Foreground Selection

Foreground selection may be performed on the basis of appearance or depth. Appearance based methods are affected by shadows, whereas depth methods are imprecise because of foreground fattening (Scharstein and Szeliski, 2002). We overcome both by using a hybrid model. The appearance of each pixel is modelled by a mixture of Gaussian distributions augmented with a shadow suppression method by (Horprasert et al., 1999). We add a fourth dimension to the model to represent the variation in each pixel's depth. A computational optimization is incorporated to discard unsupported distributions (Zivkovic, 2004).

To repress noise, morphological operations are used. The resulting foreground pixels are grouped using a connected component labelling algorithm (Horn, 1986). Blobs that contain too few pixels are discarded. The data corresponding to the remaining blobs are analysed with our head detector.

3.2 Head Detection

The depth map is treated as a point cloud in which we search for clusters arranged in a sphere that is proportional in size to human heads. Any such cluster is checked against additional constraints.

Our method uses a spherical crust template as suggested by (Mikić et al., 2003). Their method works with voxel data obtained from multi camera space carving. Our stereo camera only provides a 2.5D description, wherein occlusion plays a bigger role. In addition, their method assumes exactly one person is in view defining the problem as localisation, whereas we are interested in detecting the number of heads.

3.2.1 Template Matching

The point cloud from each blob is searched for head shapes using a template. This template consists of two spherical bounds around the same centre that define the minimum and maximum head sizes that can be detected. Both are tuned to anatomically plausible values. To enforce roundness of the point cloud portion inside the template, we augment the crust template with negative regions. The locations of these regions are illustrated in Figure 1. The region within the inner bound rejects scattered or planar clouds, whereas the region outside the outer bound enforces that the round shape is not connected to other shapes as is the case with shoulders. As a result, the template will only fit point clouds that lie around an empty core.

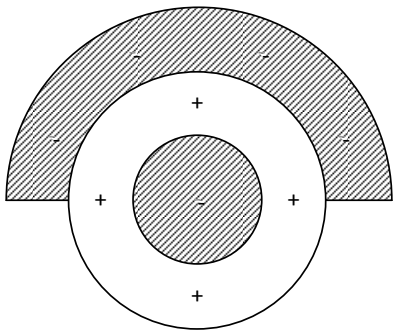
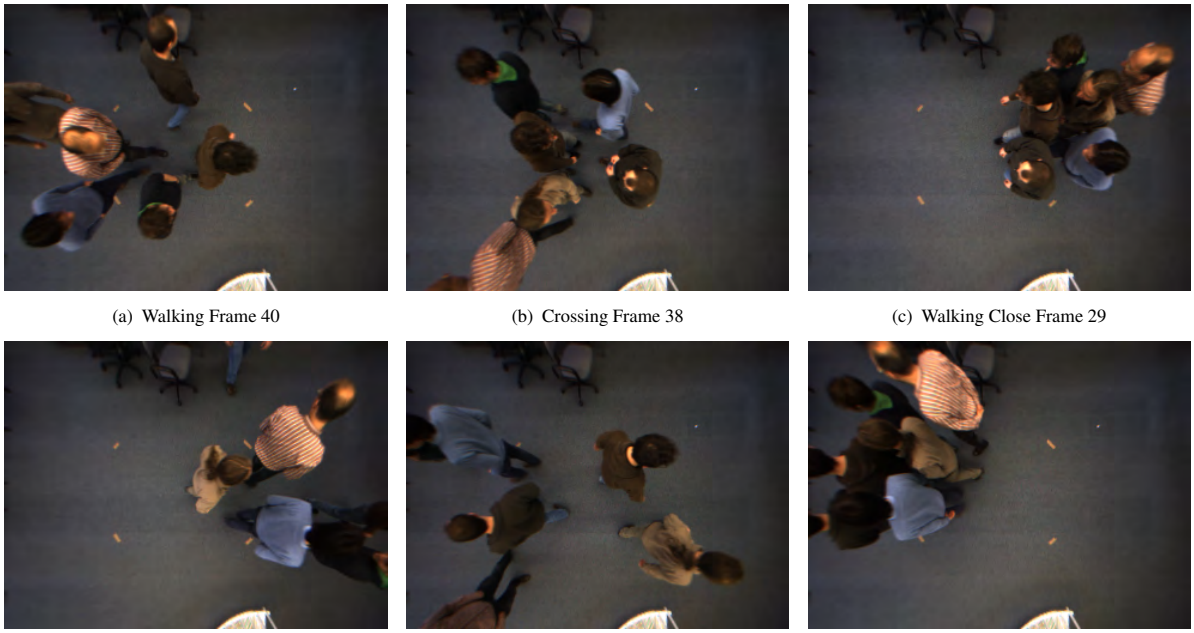


Figure 1: A cross-section of the template used for head detection showing the positive and negative regions.

Candidate heads are found by applying the template at regular discrete intervals throughout the bounding volume of a blob's point cloud. Once a head candidate is localized using this template, its exact dimensions can be calculated. A candidate is considered for further processing if it achieves a certain density and a high enough ratio between positive and negative points as described below.



(a) Walking Frame 40 (b) Crossing Frame 38 (c) Walking Close Frame 29

(d) Walking Frame 77 (e) Crossing Frame 61 (f) Walking Close Frame 64

Figure 2: (2(a), 2(d)) "Walking". (2(b), 2(e)) "Crossing". (2(c), 2(f)) "Walking close". See section 4.1 for descriptions.

3.2.2 Candidate Selection

As a first criterion for candidate selection, the number of points in the template needs to be sufficient. We can compute how many pixels the candidate would cover in the camera's view. We project a sphere with the parameters of the candidate onto the image plane and count the resulting amount of pixels.

As a second criterion, we acknowledge that the amount of noise in the depth data will cause a certain number of points to fall outside the crust and into the negative regions of the template. The number of points in the negative regions n and the number of points in the positive region p define a ratio $r = \frac{p}{p+n}$. Should r be below a particular threshold, determined by the amount of noise in the data, then the surface described by the candidate's points does not follow the template shape and the candidate is discarded.

Subsequently, the distribution of points within the template is inspected. The mean of all the candidate's points is taken to be the centre of the point cluster. The computed centre point must lie above the template's centre to prevent matches against concave curves. In addition, the points must be evenly divided around the sphere crust. This additional constraint prevents unbalanced shapes that score high only in one dense region but are not spherical.

Finally, overlapping candidates are removed. To prevent pruning a good candidate in favour of a lesser match, a fitness measure is computed according as such: $Fitness = p \cdot \frac{p}{p+n}$, where p and n are the positive and negative counts respectively. Starting with the highest scoring candidate, all overlapping candidates are discarded. This step eliminates candidates competing for the same shape. The remaining candidates are taken to correspond to heads and represent the number of people in a blob along with their positions and person heights.

3.3 Blob Separation

After head detection has been completed, all blobs that contain more than one detected head must be split. All pixels in the original blob are each assigned to exactly one new blob. The blobs are split in the range domain based on their position. First, all points in the blob as well as the centre points from the heads detected in the blob are projected on the ground plane. A Voronoi decomposition is done in this space. The ground plane projection resembles the occupation maps used by some authors. However, we already know the amount op people at this point and the projection is used for a different purpose. Without projection, a taller person in the blob would not be able to attract any points from their lower region.

Blobs in which no heads were detected can be the result of (1) a person walking into view that is not fully visible yet, (2) objects that are not people, or (3) false negative detections by the head detection algorithm. Depending on the application it can be decided

to use these blobs as if they were people, classify them as non-people or discard them.

3.4 Tracking

Producing tracks from individual detections allows route summaries to be created and detection errors to be corrected by assuming object persistence. To this end we use a set of Kalman filters. For measurements we take the projected head locations. Because we mean this method to work online, we do not look ahead and do not re-evaluate previous measurement-to-track assignments. We use the averaged Mahalanobis distance given by Equation 1 for data association, where $\vec{m}$ is a measurement, $\hat{t}$ is a track's extrapolated state and σ denotes the covariance matrix. Measurements and tracks are matched according to their smallest distance, with each measurement being matched to at most one track and vice versa.

New tracks are created for all measurements that could not be matched. Tracks that do not get any measurements assigned to them will be maintained for a set amount of time after which they are removed from the active set of tracks. In case a deactivated track has had too few measurement assignments it is likely the result of noise and is discarded.

$$D(\vec{m},\hat{t}) = \frac{\sqrt{\frac{(\vec{m}-\hat{t})^2}{\sigma(\vec{m})^2}} + \sqrt{\frac{(\vec{m}-\hat{t})^2}{\sigma(\hat{t})^2}}}{2} \quad (1)$$

Equation 1: Distance measure between an object and an extrapolated track state.

4 EXPERIMENTS

This section discusses experiments that evaluate our method. We first describe the experimental setup and the utilised data (Subsection 4.1). Subsequently, we present the experimental results (Subsection 4.2).

4.1 Experimental Setup

People counting is evaluated using three methods. All three methods are evaluated on their own and in combination with tracking. The first method is the reference and is as described in Section 3.

The second method focusses solely on blob detection done by connected component labelling, where the amount of blobs is judged against the number of visible people. Blob detection is meant to separate disconnected portions of the foreground. This alternative method works well to separate people when they do not touch or overlap. However, in crowded situations they often do (Figure 2(b) and Figure 2(d)).

The third method imposes an extra constraint on the foreground pixels before connected component labelling is applied, demanding a pixel's height to be 1.70m or higher. The effect of this is that the lower points that are usually responsible for overlap are cut off. If the cut-off height is chosen to lie above all visible shoulders but below all visible scalps then this method is essentially a crude head detector. The method is judged against the number of people for who a portion above the cut-off threshold is in view.

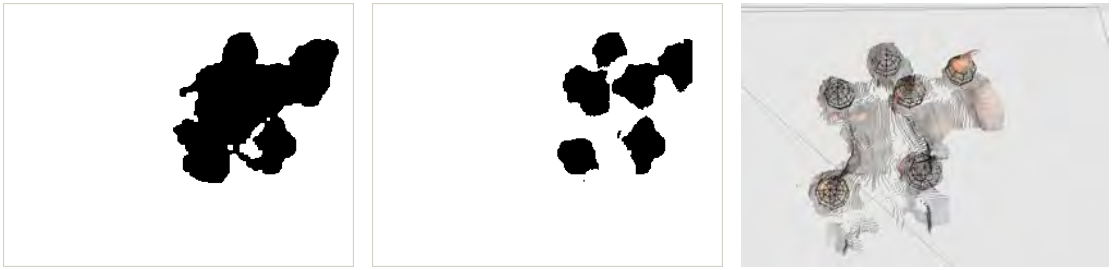
Evaluation. Each of the three methods is validated against the correct number of people that are visible to that method. To test their performances, three image sequences totalling 292 frames (Figure 2) were analysed using each of the methods. The first sequence ('walking', 106 frames) shows a group of 6 people walking across the image with all frames showing at least five people in full or partially. The second sequence ('Crossing', 85 frames) shows a group of 2 and a group of 4 people walking in opposite directions and crossing in the middle. The third sequence ('Walking Close', 101 frames) shows a group of 6 people walking across the image but closer together than in sequence 1 and staying together. These sequences present increasing amounts of overlap and mimic different real life scenarios. Ground truths were set manually in all frames for the counts that each method should have been able to reach in them.

We measure precision (the portion of results that were heads/people) and recall (the portion of heads/people that were correctly detected). Any time a head is not detected this is counted as a false negative. When a head is detected in a place where there is none this is counted as a false positive. If a head is detected but it is off target then it is counted as both a false negative and a false positive, since in this situation the head was not detected while a result was returned that was not a head.

Tracking. We evaluate precision and recall when used in combination with a tracker. In this case the tracker's position estimate is used when no head is detected at or near that location. If the estimate appears over a person's locations then it is counted as a true positive. If the tracker drifts or the person changes course and no longer appears at the extrapolated location then the tracker's estimate is counted as both a false positive and a false negative. No extra penalty was given in terms of precision and recall to tracks that switched target.

Table 1: Numeric precision and recall per sequence per method, in parentheses with tracking.

	Head Detection		*Blob Detection*		*Height Threshold*	
	Precision	Recall	Precision	Recall	Precision	Recall
All Frames	.92 (.94)	.89 (.97)	.47 (.47)	.19 (.19)	.91 (.92)	.82 (.86)
Walking	.91 (.92)	.90 (.98)	.48 (.53)	.18 (.20)	.98 (1.0)	.98 (1.0)
Crossing	.90 (.93)	.89 (.96)	.61 (.58)	.37 (.36)	.97 (.97)	.94 (.97)
Walk Close	.95 (.97)	.88 (.96)	.06 (.09)	.01 (.02)	.75 (.77)	.55 (.63)



(a) Blob detection: 1 blob (b) Height thresholding: 5 blobs (c) Head detection: 6 heads

Figure 3: A visual example of the output of the three algorithms for frame 29 of "Walking close" (Figure 2(c)).

4.2 Results

In Table 1 the results for the precision and recall experiment without tracking are given without parenthesis. We observe that without tracking, in all scenarios, the performances for both head detection and height thresholding are considerably higher than that of blob detection. Moreover, we observe that head detection consistently achieves a comparatively good performance throughout all scenarios, whereas height thresholding performs better in the walking and crossing sequences but considerably worse in the walking close scenario.

The results for the tracking enhanced experiment are shown in parenthesis in Table 1. We observe that all results are equal or better than the results without tracking except for blob detection in the crossing sequence. Further we see that in cases where results improve, recall benefits most from the addition of tracking. In only one scenario did tracks switch targets, namely in the walking close sequence using the height thresholding method. Over the course of the longest streak of misdetections 5 trackers switched targets 6 times.

Illustrations from the segmentation output and head localisation are given in Figure 3. The output for the head detector is presented in 3D with wire-frame spheres indicating the locations of heads.

5 DISCUSSION

Although height thresholding performs well under certain circumstances (walking and crossing sequences), its performance drops for crowded scenes. In the walking close sequence, stereo fattening bridges the small gap between people even at head level. Additionally, an optimal height threshold may not exist, since the difference in person height can easily put one person's scalp below another person's shoulders. Raising the threshold will overlook short people, whereas lowering it will push the performance towards that of blob detection.

As expected the addition of tracking means many false negatives can be recovered resulting in higher recall overall. However, a tracker can only increase performance if misdetections are spread out. Although the tracker can also isolate and discard single false positives, tracks that should have been finalised sometimes lingered because a false positive appeared at its extrapolated location.

The method is not limited to video sequences. Indeed, heads are detected on a per frame basis, thus the method works comparatively well on isolated frames. In the experiments described in Section 4, ground truths were created manually for all 292 frames of test data, the majority of which featured multiple heads, giving a total of 764 possible individual detections over which the results were gathered. To ensure adequate test data, the recorded sequences vary the distribution, density and appearance of people in view.

6 CONCLUSIONS AND FUTURE WORK

Experimental results show that the proposed method performs well in terms of both precision and recall. In addition, the method was shown to perform well in highly crowded situations. From our results, we may conclude that the proposed method provides a strong basis for head detection in applications that utilise stereo cameras and that it works well both on its own and in combination with tracking.

For future work, we will investigate how the blob splitting method can be enhanced for a better segmentation of people in stereo images. The detected head locations may serve as a basis for detecting other body parts and, ultimately, recognition of complete poses. Finally, given the blob splitting that the method enables, we can build rich profiles based on colour and height that can be used for cross camera correlation in a sparse camera setup.

ACKNOWLEDGEMENTS

The research reported in this paper was supported by the Foundation Innovation Alliance (SIA - Stichting Innovatie Alliantie) with funding from the Dutch Ministry of Education, Culture and Science (OCW), in the framework of the 'Mens voor de Lens' project.

REFERENCES

Beymer, D. (2000). Person counting using stereo. In *Workshop on Human Motion*, pages 127–133.

Darrell, T., Gordon, G., Harville, M., and Woodfill, J. (2000). Integrated person tracking using stereo, color, and pattern detection. *International Journal of Computer Vision*, 37(2):175–185.

Fu, H. C., Chen, J. R., and Pao, H. T. (2007). Remote head counting and tracking in crowded scene via WWW/Internet. In *Proceedings of the IADIS International Conference WWW/Internet 2007*.

Hayashi, K., Hashimoto, M., Sumi, K., Sasakawa, K., Center, A. T., Co, M. E., and Hyogo, J. (2004). Multiple-person tracker with a fixed slanting stereo camera. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings*, pages 681–686.

Heisele, B. and Woehler, C. (1998). Motion-based recognition of pedestrians. In *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, volume 2.

Horn, B. (1986). *Robot vision*. McGraw-Hill Higher Education.

Horprasert, T., Harwood, D., and Davis, L. S. (1999). A statistical approach for real-time robust background subtraction and shadow detection. In *IEEE ICCV*, volume 99.

Hoshino, T. and Izumi, T. (2006). Improvement of head extraction for height measurement by combination of sphere matching and optical flow. In *SICE-ICASE, 2006. International Joint Conference*, pages 1607–1612.

Huang, X., Li, L., and Sim, T. (2004). Stereo-based human head detection from crowd scenes. In *Proceedings of International Conference on Image Processing*, pages 1353–1356.

Ishii, Y., Hongo, H., Yamamoto, K., and Niwa, Y. (2004). Face and head detection for a real-time surveillance system. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 298–301. IEEE.

Kelly, P., O'Connor, N. E., and Smeaton, A. F. (2009). Robust pedestrian detection and tracking in crowded scenes. *Image and Vision Computing*, 27(10):1445–1458.

Luo, R. and Guo, Y. (2001). Real-time stereo tracking of multiple moving heads. In *IEEE ICCV Workshop RATFG-RTS01*, pages 55–59.

Mikić, I., Trivedi, M., Hunter, E., and Cosman, P. (2003). Human body model acquisition and tracking using voxel data. *International Journal of Computer Vision*, 53(3):199–223.

Park, S. and Aggarwal, J. K. (2000). Head segmentation and head orientation in 3d space for pose estimation of multiple people. In *IEEE Southwest Symposium on Image Analysis and Interpretation*.

Scharstein, D. and Szeliski, R. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*, 47(1):7–42.

Stauffer, C. and Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2.

Zhao, T. and Nevatia, R. (2003). Bayesian human segmentation in crowded situations. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2.

Zivkovic, Z. (2004). Improved adaptive gaussian mixture model for background subtraction. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2.