Fast Motion Estimation on Range Image Sequences acquired with a 3-D Camera

Stephan Matzka[1,2] Yvan R. Petillot[1] Andrew M. Wallace[1]

[1] Heriot-Watt University, School of Engineering and Physical Sciences, Edinburgh, UK

[2] Ingolstadt University of Applied Sciences, Institute for Applied Research, Ingolstadt, Germany

Abstract

This paper presents a computationally efficient approach to estimate translational 3-D motion from range images sequences, that is adapted from a 2-D motion estimation algorithm. An implementation of the algorithm is evaluated for computational efficiency as well as robustness in the presence of noise for both synthetic and real-life range data acquired with a PMD device, a high-speed low-resolution 3-D camera.

1 Introduction

Motion estimation for intensity video images is well researched, with a number of proven concepts to create dense motion vector fields, possessing computational efficiency, or robustness against sensor noise. However, 3-D motion estimation on range images lacks fast and robust algorithmic concepts.

A current application field for translational 3-D motion estimation is given by the use of 3-D cameras in cars, such as a PMD camera [3]. In road traffic scenes, the only notable rotational motions are yaw movements which occur for cars bending off. Yet, even in this case, translational motion dominates due to the considerable turn radius of cars.

This paper presents a novel method to estimate translational 3-D motion from range images, that is adapted from a high-performance 2-D motion estimation algorithm. Its central qualities are computational efficiency and robustness in the presence of noise.

2 Related Work

The issue of estimating 3-D motion or optical flow fields from range images has been the subject of a number of publications. For example, an evaluation of 3-D motion estimation algorithms was given in Eggert et al., 1997 [2]. Many 3-D motion estimation approaches are based upon finding correspondences. These correspondences can be considered both local (cf. [1]), or global by solving a total least squares framework [8]. The resulting flow field of the latter method is dense, yet the complexity is high and real-time computation is not feasible with current hardware.

A correspondenceless approach was pursued by Liu and Rodrigues, 1999, based upon the cross matrix to estimate the motion parameters [6]. It is also possible to use the shift of previously segmented surfaces in a range image for motion estimation [5]. This approach is restricted to small relative motion between the camera and the scene and the segmentation process in itself is complex.

Apart from the cited work on 3-D motion estimation – which is only a selection – 2-D optical flow is a major topic of interest. Most of the 2-D motion estimation algorithms used in video-encoders are designed to be computationally efficient, which is also a constraint for real-time motion estimation.

However, to estimate 3-D motion in range images under real-time constraints, neither 2-D motion estimation based on difference measures nor 3-D motion estimation algorithms with high complexity can be used. Therefore, this paper suggests adapting a 2-D motion estimation algorithm for use on range images.

3 2-D Motion Estimation using PMVFAST

The Predictive Motion Vector Field Adaptive Search Technique (PMVFAST) is a block based motion estimation technique based upon MVFAST [4], which is an essential part of several video-coding standards, such as MPEG-1/2/4 [9]. PMVFAST has shown to be faster than other motion estimators while retaining a motion estimation quality comparable to a significantly slower full search algorithm.

PMVFAST uses a Diamond Search (DS) pattern (cf. Fig. 1a). Beginning in the centre, the (0,0) motion vector (MV) is the initial starting point. Then the search path is meandering circularly around the centre, performing a full orbit each time before increasing its search distance up until the maximum search distance.

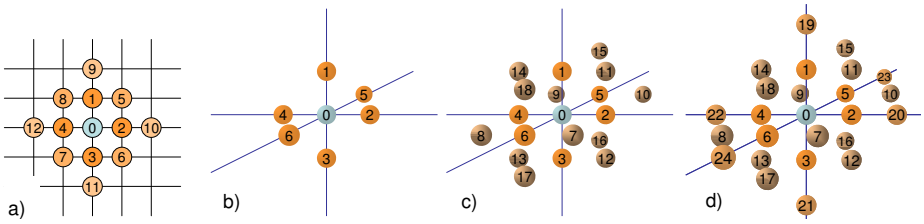

Figure 1: Fig. a) shows a Diamond Search pattern used for 2-D motion estimation. Exemplary PCS path building process: b) shows all (1,0,0) variations (#1-6), b) extends a) with all (1,1,0) variations (#7-18), and c) extends b) with all (2,0,0) variations (#19-24).

At each point on the search path, a block in the previous frame is matched against a block in the current frame. The block in the current frame is shifted by the (i,j) values of the search path. The quality of the match is determined by a distortion measure. A widely used distortion measure is the sum of absolute differences (SAD, see Eq. 1), which omits the multiplications necessary for mean squared error but has a similar performance [9]. Blocks used in this paper are 5×5 pixels, resulting in 25 summations per comparison. Also, MVs are not calculated for every pixel, instead a regular grid is used with the grid distance increasing logarithmically with the range image's size.

$$SAD_{DS}(v_x, v_y) = \sum_{i,j \in DS} \left| I_k(x+i, y+j) - I_{k-1}(x+v_x+i, y+v_y+j) \right| \qquad (1)$$

The search for the minimum SAD is performed with two differently sized diamonds in [9]. The expected magnitude of motion is estimated by examining three neighbouring

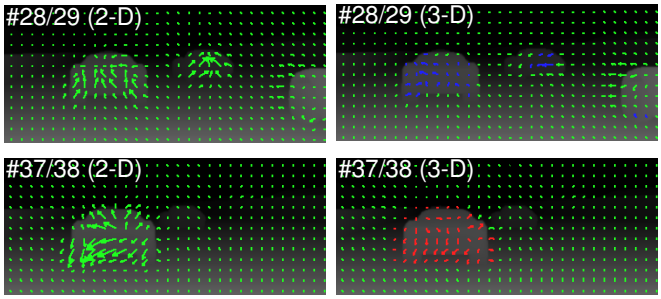

Figure 2: Motion vector fields of frame pairs #28/29 (increasing distance to car in front) and frame pair #37/38 (decreasing distance) in the Torcs sequence. Motion vector field (2D) shows the result using a 2-D full search algorithm, whereas (3D) shows the PCS result. Blue arrows indicate an increasing distance, red arrows a decreasing distance. The background shows the range images on which the motion estimation has been performed.

MVs at $(x-1,y)$, $(x,y-1)$, $(x+1,y-1)$, the previous MV at (x_{k-1},y_{k-1}), and the median MV. The average of these MVs is then used as an estimation for the current MV.

If the estimated MV for (x,y) is small, a small search diamond is used with the (0,0) MV as its centre. If the MV is estimated to be intermediate, a large diamond is used, again with (0,0) MV as starting point. In the case of high estimated motion, the small diamond is used with the estimated MV as its centre.

Regardless of the estimated motion, the (0,0) MV is examined first, and – if the distortion is below a chosen threshold – no further matching is done. Otherwise, the DS is performed and the displacement featuring the minimum distortion is chosen as the centre point in the next cycle. The search algorithm terminates if the centre of the search diamond is also the displacement with minimum distortion.

This concept holds for intensity images, yet in range images distance information is represented by intensity. On convex surfaces, such as a sphere, this will induce a difference-based 2-D motion estimation to detect a concentric outward motion if the distance is decreasing (it is implied, that small distances are represented by a high intensity), and a converging motion if the distance is increasing.

The above behaviour does not heavily affect MPEG motion estimation, since the aim there is not to calculate exact MVs, but to maximally reduce the video's bit rate while having as little visible quality loss as possible. However, for range images this effect leads to the necessity to consider depth motion in order to get accurate motion vectors.

4 Extending Diamond Search for use on Range Images

The idea of using a diamond shaped search path is extendible towards a 3-D translational motion estimation from range images. The least complex diamond shape in 3-D is a regular octahedron which will be referred to as *Point Cut Search* (PCS) path.

The PCS path will expand continuously, adding new layers around the origin in a point cut shape. The first layer has a distance of 1 to (0,0,0) and consists of the six permutations (1,0,0), (0,1,0), (0,0,1), (-1,0,0), (0,-1,0), and (0,0,-1) with varying signs.

The following base coordinates are (1,1,0); (1,1,1); (2,0,0); (2,1,0); (3,0,0) etc. These base coordinates are then permutated (maximum 6 permutations if all values are unique) with changing signs for every value (maximum 8 sign combinations if no value is zero). An illustration of the PCS path building process is given in Fig. 1b-d.

Both PMVFAST and PCS realise horizontal and vertical displacements by shifting the observation window in the actual frame horizontally and vertically. In PCS, displacements in distance in range images are represented as changes of intensity. Therefore, by adding or subtracting the value corresponding to the range displacement to the intensity values in the observation window, a displacement in distance can be modelled (see Eq. 2).

$$SAD_{PCS}(v_x, v_y, v_z) = \sum_{i,j,k \in PCS} \left| I_k(x+i, y+j) - I_{k-1}(x+v_x+i, y+v_y+j) + v_z + k \right| \quad (2)$$

As for PMVFAST, the search terminates when the centre point of the PCS is the point with minimum distortion or when the maximum number of iterations is reached.

5 Evaluation of the implemented Motion Estimator

The proposed motion estimator was implemented using four layers of abstraction (cf. Fig. 3). First, the range images are filtered in order to remove noise (temporal filtering using previous frames is optional). Second, subsequent filtered range images are searched for correspondences, using PCS. The resulting motion vectors are then filtered to remove outliers. Finally, the filtered motion vector field is used by PCS to predict the motion vectors for the next motion estimation.

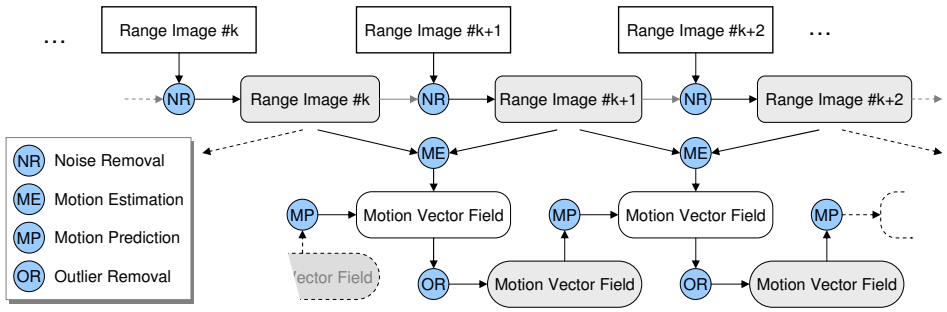

Figure 3: Block diagram of the implemented PCS motion estimator. Circles represent processing / filtering operations that are performed by the motion estimator, while boxes represent different abstraction layers from unfiltered range images to filtered MV fields.

5.1 Computational cost

The computational cost of the implemented motion estimator is evaluated using the simulated range image sequence extracted from *Torcs*[1]. The sequence consists of 155 frames

[1]Torcs is an open source racing game (`http://torcs.sourceforge.net`) using OpenGL. See supplementary video: `http://emfs1.eps.hw.ac.uk/~ceeyrp/BMVC2007/motionTorcs.avi` showing the range im-

recorded at 15 frames per second and a resolution of 500×220 pixel. Range is encoded with 8 bit, representing 256 range values, which is a coarse yet sufficient range resolution.

For each configuration, the average number of comparisons needed for each motion vector and the average SAD for the chosen motion vectors were taken as indicators of the computational cost and motion vector field quality respectively. To get a benchmark for these two values, a Full Search (FS) has been used (cf. Tab. 1).

Evaluating the performance of the PCS search strategy, the maximum number of iterations to shift the local minimum to the PCS's centre is the most important parameter. For evaluation, two PCS paths were chosen. The small PCS used has a maximum search distance of 2, the large PCS has a maximum search distance of 5. The threshold for (0,0,0) MV was set to 16, a value which yielded good results in the evaluation.

	FS	PCS_2	PCS_3	PCS_4	PCS_5	PCS_6	PCS_7
Comparisons per MV	75.52	19.72	22.49	24.70	25.72	26.48	27.10
Average SAD per MV	33.16	45.46	40.21	37.04	35.69	34.60	33.86
Efficiency Measure (Product)	2504.4	897.5	904.3	915.0	918.0	916.1	917.6

Table 1: Comparisons per MV and average SAD for motion estimation in the Torcs sequence. A full search (FS) is used as benchmark for the PCS_n with n maximum iterations. The efficiency measure is the product of comparisons per MV and average SAD.

Tab. 1 shows the performance of the PCS strategy with respect to the maximum number of iterations allowed. The lowest average SAD of 33.86 for 7 maximum iterations comes very close to the benchmark value $\overline{SAD}_{full}$ of 33.16, while needing only about a third (36%) of the comparisons.

In order to assess the efficiency of the PMVFAST search strategy, the product of comparisons needed for each MV and the average SAD is a possible metric. This product grows with increasing computational cost and distortion, for low computational cost and low distortion the product is small (cf. Tab. 1), the latter being true for PCS.

5.2 Quantitative Evaluation of Accuracy

A comparison of the estimated motion vector fields of a synthetic motion pattern against a ground truth known from the rendering process of the pattern has been conducted.

5.2.1 Motion Ground Truth

The motion pattern consists of two spheres diametrically orbiting around the range image's centre (x,y,z) = (160,120,127) so that the sphere in front occludes the sphere behind it intermittently. The underlying motion function for this pattern is

$$v_k = \begin{pmatrix} \lfloor 80.0 \cdot \sin(\frac{k}{30}) + 160.5 \rfloor \\ \lfloor 60.0 \cdot \cos(\frac{k}{30}) + 120.5 \rfloor \\ \lfloor 80.0 \cdot \cos(\frac{k}{30}) + 127.5 \rfloor \end{pmatrix} - \begin{pmatrix} x_{k-1} \\ y_{k-1} \\ z_{k-1} \end{pmatrix}, \; v_{max} = \begin{pmatrix} 3 \\ 2 \\ 3 \end{pmatrix} \tag{3}$$

The resulting range image sequence contains 200 frames with 320×240 pixels[2].

age, and the motion vector field estimated using PCS.

[2]See supplementary video: `http://emfs1.eps.hw.ac.uk/~ceeyrp/BMVC2007/motionOrbit.avi` showing the source range image, ground truth, motion estimation and motion vector field (from left to right).

5.2.2 Noise and Preprocessing Model

Range data sequences acquired by a 3-D camera suffer from a substantial amount of noise. This noise can be reduced by employing temporal filtering of a large number of frames. For traffic scenes, temporal filtering over a number of frames increases rotational motion of other traffic participants, which is not handled well by the algorithm.

In this trade-off between noise and rotational motion the algorithm has shown to be more capable of handling noise in the range images, therefore a diminutive number of frames for temporal filtering has been chosen.

The noise that occurs in 3-D camera range data sequences is best characterised as clipped Gaussian noise, as no negative distances or distances above the maximum measurable distance can appear, yet the distribution of noise suggests a Gaussian distribution (cf. Fig. 4). Therefore, the synthetic range image sequence has added noise of Gaussian distribution, where $0.0 \geq z(x,y) + z_{noise} \geq 255.0$.

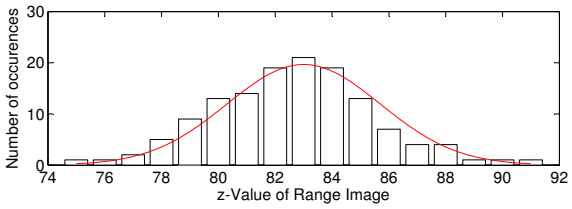

Figure 4: Distribution of range measurements of a constant distance over 135 frames (bars), which can be approximated by a Gaussian distribution with σ = 2.7 (red line).

Assuming a Gaussian noise model, spatial filtering using a Gaussian filter with $0.8 \geq \sigma_{RI} \geq 4.8$ presents a suitable preprocessing (cf. Fig. 5).

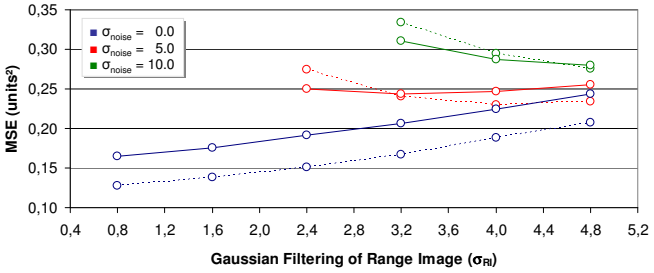

Figure 5: MSE of motion vector components for the orbiting movement pattern under influence of Gaussian noise σ_{noise} estimated by PCS_3 (solid line) and FS (dotted line) as compared to ground truth. The range image is processed using a Gaussian filter with σ_{RI}.

In Fig. 5, three major effects can be observed. First, if a noise-free range image is processed with a Gaussian filter, the MSE deteriorates as could be expected. Second, if a noisy range image is processed with a Gaussian filter, the MSE improves until a point where the range image is quasi noise-free and then shows the same behaviour as a noise-free image, that is MSE deterioration for higher standard deviations.

The third observable effect is, that PCS has a lower MSE for range images with a high

remaining noise after preprocessing. The reason for that is differing termination conditions. If a high level of noise is present during motion estimation, the correct MV does not necessarily exhibit the lowest SAD value. Using a full search, every displacement has the same probability to be selected as the estimated MV, whereas the iterative shifting in PCS makes it more probable, that a displacement near the initial starting point is selected.

The synthetic scene contains a large fraction of (0,0,0) MVs, therefore an incorrect MV close to an initial (0,0,0) MV starting point does not affect the MSE as much as a large MV, that is more probable to occur using a full search. However, it can be seen in Fig. 5 that this effect disappears when a suitable level of filtering is applied, so that the correct MSE exhibits the minimum SAD.

5.2.3 Regularisation Model

An analysis of the resulting MV fields against the ground truth suggest, that the main reason for high MSE values of the estimated motion vector fields is outliers caused by noise in the range image, not generic false motion vector estimation. Suitable methods to achieve outlier reduction include Gaussian or median filtering of the MV field.

In Fig. 6, MSE values for the same synthetic range image sequence as in Fig. 5 when using a Gaussian ($\times$) or median (Δ, using a 5×5 field) filter are shown.

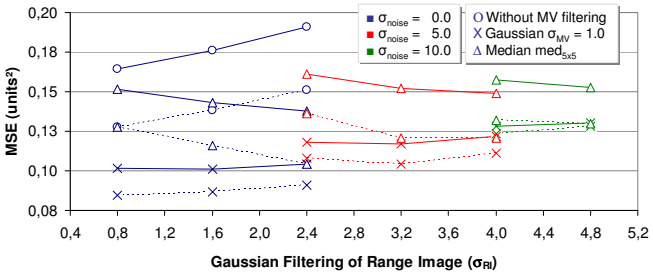

Figure 6: MSE of motion vector components for the orbiting movement pattern under influence of Gaussian noise σ_{noise} estimated by PCS_3 (solid line) and FS (dotted line) as compared to ground truth. The source range image is filtered using a Gaussian filter with σ_{RI}. The motion vector field is filtered using a Gaussian ($\times$) or median (Δ) filter.

In can be seen from Fig. 6, that the optimum MSE values gained by PCS at different levels of noise in the range images (including no noise) are within a narrow field (that is 0.1015 to 0.1616). Thich is an indicator that the algorithm is robust towards noise, if both input range images and motion vector fields are suitably filtered. The results are also comparable with the results gained by FS. At the same time, PCS computed the 320×240 pixels range image sequence at 11.8 frames per second (fps) on a standard 2.0 GHz PC, where FS performed at 1.85 fps, thus being more than six times (6.38) slower.

5.3 Performance on Data acquired with a 3-D Camera

In addition to synthetic range image sequences, the proposed algorithm has been evaluated using real-life data acquired by a PMD device, a high-speed low-resolution 3-D camera.

The 3-D camera is mounted inside the car close to the rear-mirror, observing an angle range of $55^\circ \times 18^\circ$ in front of the car. It acquires 64×16 pixel range images for distances up to 30m with a frame-rate of $\geq$100Hz [3]. In order to acquire a ground truth, a 2-D laser-scanner mounted on the car's radiator grille was used (see Fig. 7).

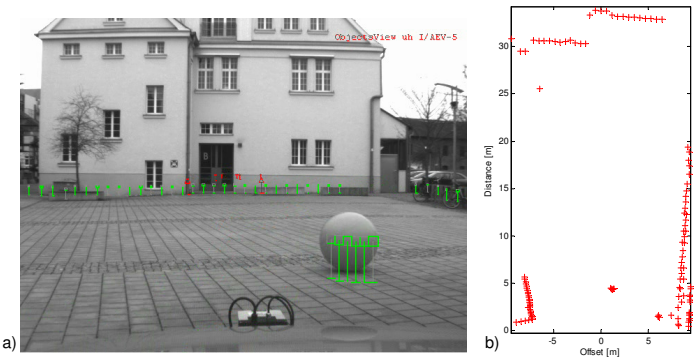

Figure 7: The left image a) shows the scene at frame #310 as seen from a grayscale intensity camera mounted close to the PMD device. The scatterplot b) on the right side shows the readings of the 2-D laser-scanner at the same frame.

As the proposed algorithm is designed to estimate translational motion, a large rubber ball is used due to its rotational invariance. Moreover, it is possible to reconstruct the ball's 3-D shape from the measured 2-D scan-line at any time, as the ball's radius and the scan-line's height are both known. In the scene, the ball is pushed in front of the stationary car and – due to a slightly inclined ground plane – performing a curve to the left, heading back towards the car (cf. Fig. 8a).

In order to determine the trajectory of the ball's centre, the readings of the laser-scanner are discarded unless they fall into a rectangle (distance 0..15m, offset -5..5m), which exclusively returns readings showing the ball. These readings fall onto a circle with the ball's radius. Thereby the ball's centre is determined fulfilling the circle equation Eq. 4 for the selected laser readings $(x_{reading}, y_{reading})$.

$$x_{centre}, y_{centre} = \arg \left(x_{reading_{1,2,\ldots,n}} - x_{centre}\right)^2 + \left(y_{reading_{1,2,\ldots,n}} - y_{centre}\right)^2 \qquad (4)$$

It is obvious, that Eq. 4 is overdetermined for $n > 2$, which can be solved by averaging all centre positions which are calculated using 2 laser readings at a time. The centre positions are then processed by applying both median and Gaussian filters in order to get a continuous motion (see Fig. 8a).

The range image sequence of the same scene is acquired with a PMD device[3] (see Fig. 8b). In order to be used with PCS, the range data has to be filtered over a small number of frames and outliers have to be rejected. Spatial filtering is not performed at this point, as the motion estimation algorithm includes this operation.

Generating a motion ground truth from the laser readings requires a calibration function from (x_{laser}, y_{laser}) to $(x_{pmd}, y_{pmd}, z_{pmd})$, which is approximated using a L_2 regression. (cf. [7]).

[3]See supplementary video `http://emfs1.eps.hw.ac.uk/~ceeyrp/BMVC2007/motionPMD.avi` showing the source range image, ground truth, motion estimation and motion vector field (from top to bottom).

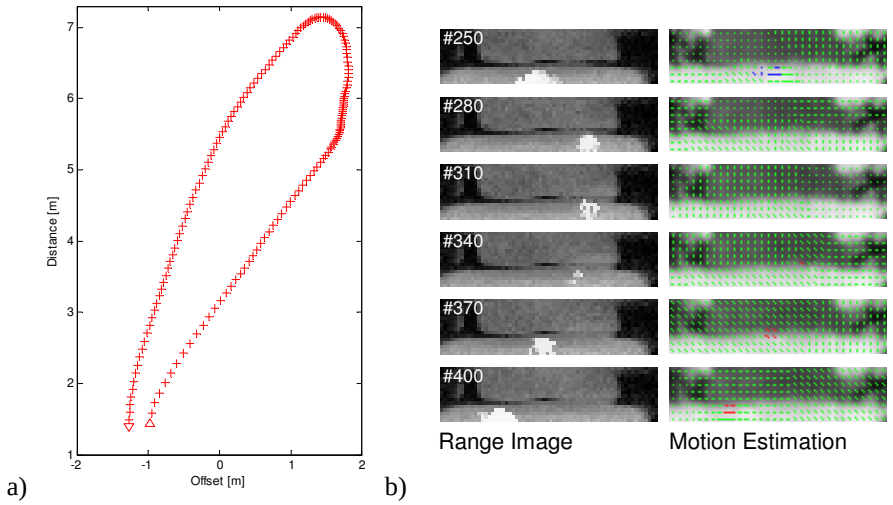

Figure 8: Scatterplot a) shows the ball's trajectory as detected with a laser scanner (Δ represents frame #250, ∇ frame #400). The range image sequence b) shows selected frames of the scene as seen by the PMD device (ball is brightened manually as to enhance visibility in the range image) as well as the corresponding estimated motion vector field. In the latter, blue arrows indicate an increasing distance, red arrows a decreasing distance.

The motion ground truth can now be generated from the ball's centre position. In Fig. 9 the MSE values of the motion estimation for the acquired range image sequence as compared to ground truth are shown.

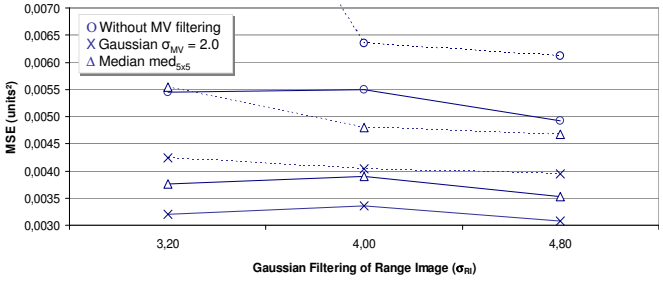

Figure 9: MSE of motion vectors components estimated by PCS_3 (solid line) and FS (dotted line) as compared to the ground truth under influence of Gaussian noise σ_{noise} for the orbiting movement pattern. The source range image is processed using a Gaussian filter with σ_{RI}.

Fig. 9 shows, that Gaussian or median filtering of the motion vector field results in a considerable reduction of the MSE. Both PCS and FS show small MSE values. Due to the large fraction of (0,0,0) MVs in the ground truth, the FS suffers from normal distribution of incorrect MVs in the presence of unfiltered noise, which is discussed in section 5.2.2 above. Again, PCS (46.9 fps) performed significantly faster than FS (19.5 fps) at a comparable motion vector quality.

6 Conclusion and Future Work

This paper presented a novel method to efficiently determine 3-D translational motion vectors in a range image sequence. The motion estimation has been evaluated on noisy, synthetic, and real-live range image data acquired by a PMD device and shown to be robust if a suitable filtering is applied on both range image and motion vector field.

Yet, there remain limitations for the proposed algorithm, which are largely those of PMVFAST. First, occlusion boundaries with little contrast between foreground object and background can lead to a motion vector pointing from the previous scene's background towards the occluding object's surface and vice versa. Second, rotational movements of objects must not be fast in order to find correct correspondences, which is generally true when using a high-speed 3-D camera on a road traffic scene. However, there still exists a trade-off between rotational motion and noise in range image sequences.

It has been shown that the computational cost for the acquisition of the motion vectors is low when compared to a full search. At a comparable motion vector field quality, PCS is shown to require only 16% – 42% of the number of comparisons a full search performs.

Future work will include evaluating the algorithm allowing a dynamic road-traffic range image scene as opposed to a static background and a fixed camera position. We should also evaluate other alternatives to the full search algorithm such as range flow, phase correlation or the use of a correlation-based matching criterion instead of a difference-based SAD measure.

References

[1] Krishnendu Chaudhury, Rajiv Mehrotra, and Cid Srinivasan. Detecting 3-d motion field from range image sequences. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 29(2):308–314, 1999.

[2] D. Eggert, A. Lorusso, and R.B. Fisher. Estimating 3-d rigid body transformations: a comparison of four major algorithms. *Machine Vision and Applic.*, 9:272–290, 1997.

[3] B. Fardi, J. Dousa, G. Wanielik, B. Elias, and A. Barke. Obstacle detection and pedestrian recognition using a 3d PMD camera. In *IEEE Intelligent Vehicles Symposium*, 2006.

[4] P.I. Hosur and K.K. Ma. Motion vector field adaptive fast motion estimation. In *Second International Conference on Information, Communications and Signal Processing*, 1999.

[5] X. Jiang, S. Hofer, T. Stahs, I. Ahrns, and H. Bunke. Extraction and tracking of surfaces in range image sequences. In *Proceedings of the 2nd International Conference on 3-D Digital Imaging and Modeling*, pages 252–260, 1999.

[6] Yonghuai Liu and Marcos A. Rodrigues. Correspondenceless motion estimation from range images. In *Proceedings of the Seventh International Conference on Computer Vision (ICCV'99)*, volume 1, pages 654–660. IEEE Computer Society, 1999.

[7] P. J. Rousseeuw and A. M. Leroy. *Robust regression and outlier detection*. John Wiley & Sons, Inc., New York, NY, USA, 1987.

[8] H. Spies, B. Jähne, and J. L. Barron. Range flow estimation. *Computer Vision Image Understanding*, 85(3):209–231, 2002.

[9] Alexis Michael Tourapis, Oscar C. Au, and Ming Lei Liou. Predictive motion vector field adaptive search technique (PMVFAST) - enhancing block based motion estimation. In *Proceedings of Visual Communications and Image Processing*, 2001.