FAST DEPTH-INTEGRATED 3D MOTION ESTIMATION AND VISUALIZATION FOR AN ACTIVE VISION SYSTEM

M. Salah E.-N. Shafik and Bärbel Mertsching
GET Lab, University of Paderborn, Pohlweg 47-49, 33098 Paderborn, Germany

Keywords: 3D Motion parameters estimation, 3D Motion segmentation, Depth from stereo, Active and robot vision.

Abstract: In this paper, we present a fast 3D motion parameter estimation approach integrating the depth information acquired by a stereo camera head mounted on a mobile robot. Afterwards, the resulting 3D motion parameters are used to generate and accurately position motion vectors of the generated depth sequence in the 3D space using the geometrical information of the stereo camera head. The proposed approach has successfully detected and estimated predefined motion patterns such as motion in the Z direction and motion vectors pointing to the robot which is very important to overcome typical problems in autonomous mobile robotic vision such as collision detection and inhibition of the ego-motion defects of a moving camera head. The output of the algorithm is part of a multi-object segmentation approach implemented in an active vision system.

1 INTRODUCTION

3D motion interpretation has evolved into one of the most challenging problems in computer vision. The process of detecting moving objects as well as the estimation of their motion parameters provides a significant source of information to better understand dynamic scenes. The motion in computer vision is related to the change of the spatio-temporal information of pixels. Computing a single 3D motion from a 2D image flow by finding the optimal coefficient values in a 2D signal transform suffers from ambiguous interpretations concerning 3D motion especially motions in the Z direction. On the other hand, one of the main challenges facing the segmentation of 3D multi-moving objects in an active vision system is the segmentation of an incoherent MVF into partitions in reasonable computation time. This especially proved to be difficult when moving objects are partially visible and not connected. Hence, it is important to detect, estimate, and segment the MVF independently from a predefined spatial coherence such as object contours generated from image segmentation approaches. Such methods are dependent on a group of features which could be affected by the continuous environment change in a dynamic scene, e.g., the results of the color-based segmentation approaches could be affected by illumination changes.

Yet, some of these 3D motion estimation and segmentation approaches require a pre-defined 3D model before the surface projection model or prior segmentation information (Schmudderich et al., 2008), which is considered a vital drawback in the autonomous robotic field where unpredicted scenarios and model geometry may exist. Moreover, they did not address the multi-moving non-rigid objects problem where several objects could be occluded in different depth levels (Kim et al., 2010). Furthermore, integrating the depth information provides accurate estimation for motion in the z direction even for a static vision system which is not applicable to monocular systems (Li et al., 2008; Ribnick et al., 2009). Another aspect that should be taken into consideration is the computation speed as active vision applications require fast algorithms to act realistic in such a dynamic environment.

In this paper, a new algorithm is proposed to enhance the computational speed of the motion segmentation approach presented in (Shafik and Mertsching, 2008) by integrating the depth information in the 3D motion parameters estimation process. Hence, the search space has been reduced to be five dimensions which represent the rotation around the $x, y,$ and z axes and translation in the direction of the $x-$ and $y-$axis. The geometrical information of the mobile robot and the mounted stereo camera head has been taken into consideration in order to accurately position the motion vectors in the 3D spatial domain. The resulting 3D MVF provide the ability to detect and estimate any predefined motion patterns which is vital for predicting any possible collision not only with the robot

Salah E.-N. Shafik M. and Mertsching B..
FAST DEPTH-INTEGRATED 3D MOTION ESTIMATION AND VISUALIZATION FOR AN ACTIVE VISION SYSTEM.
DOI: 10.5220/0003315100970103
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2011), pages 97-103
ISBN: 978-989-8425-47-8

but with any objects in the observed 3D environment. The disparity map is generated using a segment-based scan line stereo algorithm presented in (Shafik and Mertsching, 2009) which is fast and independent of the GPU power (needed for other applications). The research presented in this paper is intended to be included in active vision applications (Ali and Mertsching, 2009; Aziz and Mertsching, 2009). In order to analyze those applications in a scalable complex scene, a virtual environment for simulating a mobile robot platform (SIMORE) is used (Kotthäuser and Mertsching, 2010).

The remainder of the paper is organized as follows: section 2 gives an account of the related work to the proposed method, while section 3 describing in details the proposed algorithm. Section 4 discusses the results of experiments and evaluates the outcome of the proposed method, and finally, section 5 concludes the paper.

2 RELATED WORK

In (Massad et al., 2002; Shafik and Mertsching, 2007), a 3D motion segmentation approach is conceptually able to handle transparent motion which describes the perception of more than one velocity field in the same local region of an image despite the pixel-connectivity of objects where motion parameters are used as a homogeneity criterion for the segmentation process.Other approaches in this context assume that each segment represents a rigid and connected object such as (Gruber and Weiss, 2007) where 2D non-motion affinity cues are incorporated into 3D motion segmentation using the Expectation Maximization (EM) algorithm. In the Expectation step, the mean and covariance of the 3D motions are calculated using matrix operations, and in the Maximization step the structure and the segmentation are calculated by performing energy minimization. In (Sotelo et al., 2007) the ego-motion problem has been handled using a stereo vision system where feature points (basically road lane markings) are matched between pairs of frames and linked into 3D trajectories. However, the estimated parameter is only the vehicle velocity. Recently, many works have concentrated on the study of the geometry of dynamic scenes by modeling dynamic 3D real world objects (Rosenhahn et al., 2007; Yang and Wang, 2009) where the projected surface of a 3D object model and the data of a previously estimated 3D pose are used to generate a shape prior to the segmentation process. The goal of 2D-3D pose estimation is to estimate a rigid motion which fits a 3D object model to 2D image data. Choosing which features are used for the object model is very important to determine the 3D pose by fitting the selected feature to corresponding features in an image. In this case, the feature is the object surface with the object silhouette which implements 2D non-motion affinity cues generated from object segmentation. (Hasler et al., 2009) suggested a texture model based method for 3D pose estimation. Contour and local descriptors are used for matching, where the influence of the features is automatically adapted during tracking. This approach has shown its ability to deal with a rich textured and non-static background as it has shown robustness to shadows, occlusions, and noise in general situations overcoming the drawbacks of the single features. However, the use of several cameras from different angles is necessary for the estimation of 3D object positions which is not the case for a single mobile robot. Another application for motion segmentation and 3D modeling (Yamasaki and Aizawa, 2007) for consecutive sequences of 3D models (frames) represented as a 3D polygon mesh has conducted the motion segmentation by analyzing the degree of motion using extracted feature vectors, while each frame contains three types of data such as coordinates of vertices, connection, and color.

On the other hand, using the spatial coherence as in (Pundlik and Birchfield, 2006; Taylor et al., 2010) requires prior information of the object geometry. Such information is mainly based on a predefined assumption of spatial constraints or detecting certain groups of feature points which in the case of our autonomous system are not available. In addition, implementing such constraints leads to image segmentation rather than segmenting the generated MVF based on its motion parameters.

3 PROPOSED ALGORITHM

In this part, the functionality of the proposed algorithm will be described. In a neural system for interpreting optical flow (Tsao et al., 1991), the computation of a 3D motion from a 2D image flow or a motion template finds the optimal coefficient values in a 2D signal transform. The ideal optical motion υ_{opt} caused by motion of a point (x,y,d) on a visible surface $d = \rho(x,y)$, is

$$\upsilon_{opt}(x,y) = \sum_{i=1}^{6} c_i e_i(x,y) \qquad (1)$$

where $e_i(x,y)$ represents the six infinitesimal generators in form of a 2D vector field.

$$
\begin{aligned}
e_1(x,y) &= \begin{pmatrix} \rho^{-1}(x,y)\sqrt{1+x^2+y^2} \\ 0 \end{pmatrix} \\
e_2(x,y) &= \begin{pmatrix} 0 \\ \rho^{-1}(x,y)\sqrt{1+x^2+y^2} \end{pmatrix} \\
e_3(x,y) &= \begin{pmatrix} -x\rho^{-1}(x,y)\sqrt{1+x^2+y^2} \\ -y\rho^{-1}(x,y)\sqrt{1+x^2+y^2} \end{pmatrix}
\end{aligned} \tag{2}
$$

and for rotation :

$$
\begin{aligned}
e_4(x,y) &= \begin{pmatrix} -xy \\ 1+y^2 \end{pmatrix} \\
e_5(x,y) &= \begin{pmatrix} 1+x^2 \\ xy \end{pmatrix} \\
e_6(x,y) &= \begin{pmatrix} -y \\ x \end{pmatrix}
\end{aligned} \tag{3}
$$

Integrating the depth information into the 3D motion parameters estimation process reduces the search space to 5D where the parameter coefficient of the translation in z direction c_3 will equal the depth difference between two consecutive disparity maps:

$$c_3^i = d_i^{t+1} - d_i^t \tag{4}$$

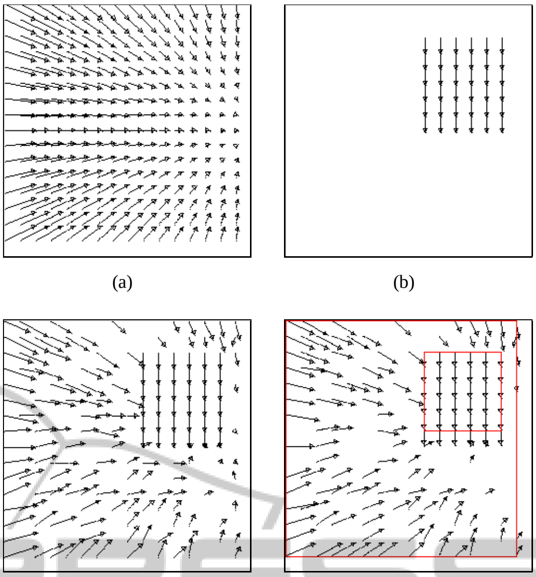
where d_i^t is the depth of point $P_i(x,y,t)$ and d_i^{t+1} is the depth of its correspondence point $P_i(x+\delta x, y+\delta y, t+1)$ determined by the motion vector $Vf(i)$ generated using a fast variational optical flow approach (Bruhn et al., 2005). Before the estimation approach starts, a noise reduction process is applied to the input MVF in order to limit the estimation process to the valid vectors only. Then, a motion segments class is initialized where every segment contains the motion parameters information $c(\xi^i)$ of the attached motion. The segmentation process considers the whole MVF representing one motion at the first iteration.

A validation process is applied to each unprocessed vector $\upsilon_k^{\varepsilon=0}, \varepsilon \in \{1,0\}$ in order to detect whether it belongs to the same motion or not by measuring the vector difference ϑf_k between the estimated vector and the actual input vector.

$$
\begin{aligned}
\vartheta f_k(p_m) &= \upsilon_k^{\varepsilon=0} - \upsilon_{inp}(p_m) \\
& \qquad \upsilon_k^{\varepsilon=0} \in \xi_i \quad if \quad \vartheta f_k(p_m) < \tau_{min}^{\vartheta f}
\end{aligned} \tag{5}
$$

where $\tau_{min}^{\vartheta f}$ is the minimum threshold that a vector difference should pass in order to consider an estimated vector $\upsilon_k^{\varepsilon=0}$ belonging to the current motion segment ξ^i generated by the motion parameters $c(\xi^i)$

For an image point p_m, the update process starts by estimating the motion parameters $c(p_m)$ using the following error function



(a)

(b)

(c)

(d)

Figure 1: Segmentation of two different synthetic motions: (a) first motion, (b) second motion, (c) noisy MVF consists of the two previous motions, (d) result of the motion segmentation approach.

$$E_k(c(p_m)) = \frac{1}{|V|} \sum_{p \in V} \sqrt{(\vartheta f_k(p_m))^2} \tag{6}$$

The estimation process is re-applied after the exclusion of vectors that do not belong to the same motion. Fig. 1 demonstrates the result of motion segmentation of two different synthetic motions.

3.1 3D Representation of Motion Parameters

The visualization difference between a projected 3D point into a 2D plane using the equations proposed in (Tsao et al., 1991) and the 3D homogeneous transformation matrix resulting from multiplying the current 3D spatial position and the perspective matrix must be taken into consideration. Hence, in order to represent a similar visualization of the projected 3D point in the real 3D spatial domain using the OpenGL libraries, transformation functions have to be applied to estimate the OpenGL transformation matrix coefficients (t_x, t_y, t_z for translation motion and $\theta_x, \theta_y, \theta_z$ for rotation motion) from the pre-estimated 3D motion parameter coefficients of the projected motion c_i (eq. no. 1). The projective transformation requires an external calibration of the camera geometry to obtain the scale information (Ribnick et al., 2009).

The translation in the x and y direction will be equal to the pre-estimated 3D motion parameters c_1, c_2, while the translation in the z direction and the rotation motions involve the perspective information of the OpenGL Frustum function. In OpenGL, a 3D point in eye space is projected onto the near plane (projection plane) using the following transformation matrix:

$$\begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = \begin{bmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{-f+n}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} x_e \\ y_e \\ z_e \\ w_e \end{bmatrix} \tag{7}$$

Hence the translation in z direction t_z will be

$$t_z = -\frac{x_e n}{x_e - c_3 x_s k} \tag{8}$$

where $x_s \in [-1,1]$ is the normalized value of the x_e location on the near plane, k is a scaling factor.

In order to estimate the rotation parameters such as the rotation about the z axis θ_z, the following transformation matrix has to be used:

$$\begin{bmatrix} x' \\ y' \\ z' \\ w \end{bmatrix} = \begin{bmatrix} cos\theta_z & -sin\theta_z & 0 & 0 \\ sin\theta_z & cos\theta_z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_e \\ y_e \\ z_e \\ 1 \end{bmatrix} \tag{9}$$

the value of x derived from eq. no. 3 will be used.

$$\begin{gathered} e_6(x,y) = \begin{pmatrix} -y \\ x \end{pmatrix} \\ y = y_s k \\ x = x_e - c_6 y_s k \end{gathered} \tag{10}$$

from eq. no. 7 and 9:

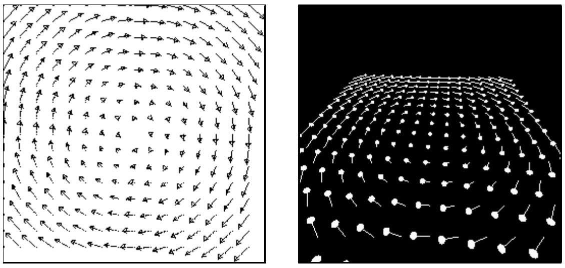
$$x = \frac{-nx'}{z'} = \frac{-n(x_e cos\theta_z - y_e sin\theta_e)}{z_e} \tag{11}$$

from eq. no. 10 and 11 :

$$\theta_z = sin^{-1}\left(\frac{(x_e - c_6.y_s.k).\frac{-z_e}{n}}{\sqrt{x_e^2 + y_e^2}}\right) - tan^{-1}\left(\frac{x_e}{-y_e}\right) \tag{12}$$

Fig. 2 demonstrates the rotation around the z axis using the rotation parameter coefficient c_6 from (eq. no. 3) and the transformed rotation parameter θ_z from (eq. no. 12).

The same procedure is applied for the estimation of the rotation parameters θ_x and θ_y:



(a) (b)

Figure 2: Rotation around z axis. (a) using the rotation parameter coefficient c_6, (b) perspective view of the transformed rotation parameter θ_z using OpenGL.

$$\theta_x = tan^{-1}\left(\frac{-y_e(n + z_e) - z_e c_4 y_s^2 k}{y_e(y_e + c_4 y_s^2 k) - nz_e}\right) \tag{13}$$

$$\theta_y = tan^{-1}\left(\frac{x_e(n + z_e) + z_e c_5 x_s^2 k}{x_e(x_e + c_5 x_s^2 k) - nz_e}\right) \tag{14}$$

3.2 3D Representation of Motion Vectors

In order to estimate the metric values of the disparity maps, the distance between the stereo cameras b and the focal length f has to be known. Stereo algorithms search only a window of disparities where the range of determined objects is restricted to some interval called the *Horopter*. The search window can be moved to an offset by shifting the stereo images along the baseline and must be large enough to encompass the ranges of objects in the scene. Hence, the determined depth value d will be:

$$d = \frac{b.f}{x_r - x_l} \tag{15}$$

where $x_r - x_l$ is the disparity value.

The representation of a vector in the 3D domain requires the 3D spatial information of its two points $P_i1(x,y,z)$ and $P_i2(x',y',z')$:

$$P_i1(x,y,z) = \begin{pmatrix} x_i \frac{d}{f} \\ y_i \frac{d}{f} \\ d_i^t \end{pmatrix} \tag{16}$$

$$P_i2(x',y',z') = \begin{pmatrix} (x_i + U_i)\frac{d}{f} \\ (y_i + V_i)\frac{d}{f} \\ d_i^{t+1} \end{pmatrix} \tag{17}$$

For an accurate 3D representation of the 2D MVs, U_i and V_i are functions of the depth information:

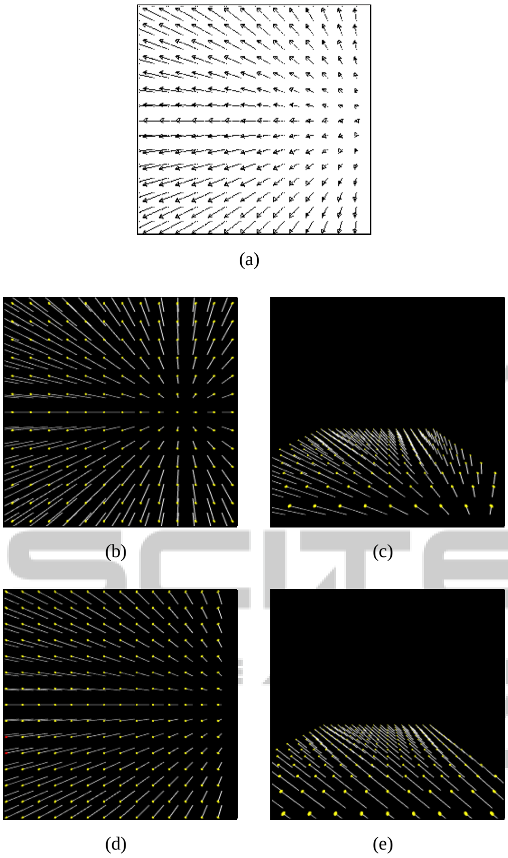


Figure 3: A synthetic 3D motion templates. (a) the generated 2D MVF of the motion parameters $c = (1,0,1,0,0,0)$ representing translation in the x and z direction. (b-c) the incorrect 3D MVF and its perspective view in OpenGL generated using u_i and v_i values of the 2D MVF. (d-e) the correct 3D MVF generated using U_i and V_i values.

$$U_i = u_i + (d_i^{t+1} - d_i^t)x_s \tag{18}$$

$$V_i = v_i + (d_i^{t+1} - d_i^t)y_s \tag{19}$$

where the u_i and v_i are the 2D generated MV components. Fig. 3 represents the error resulting from using the 2D MV components u_i and v_i in the estimation of x' and y' values of a 3D motion parameters $c = (1,0,1,0,0,0)$ representing translation in the x and z direction.

4 RESULTS AND DISCUSSION

In this section, the result of applying the proposed approach to two different data sets will be presented. In order to correctly test and analyze the result of the proposed algorithm, a virtual environment simulating a mobile robot in a scalable complex scene is used. In this environment the simulated robot is in front of a stable cube, a moving cone, and a size changeable ball. The direct 3D representation of disparity maps generated from the stereo image sequences without taking into consideration the perspective transformation results in falsely positioning the MVF in the 3D spatial domain. Fig. 4 demonstrates the error of a direct 3D representation of disparity maps where the disparity values belonging to the scene ground are falsely located along the y axis, and the result of the proposed 3D MVF representation where the MVs belong to the same scene ground are correctly positioned.

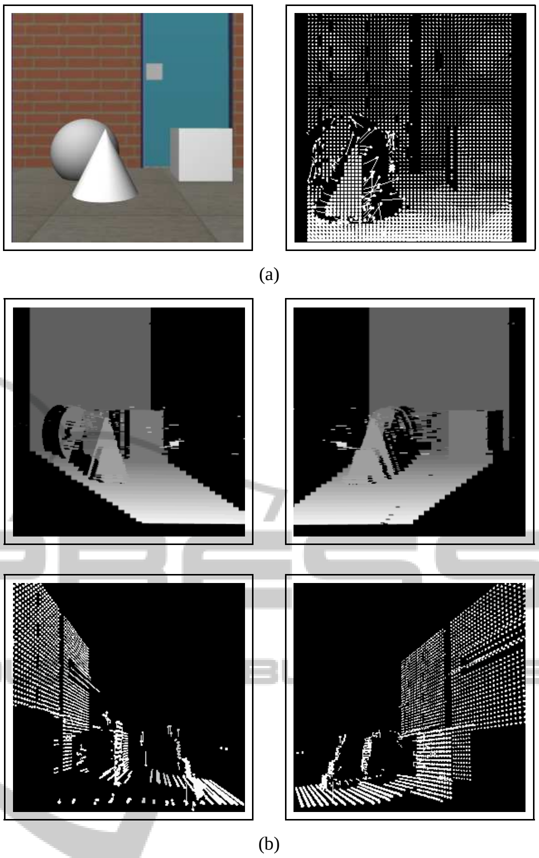
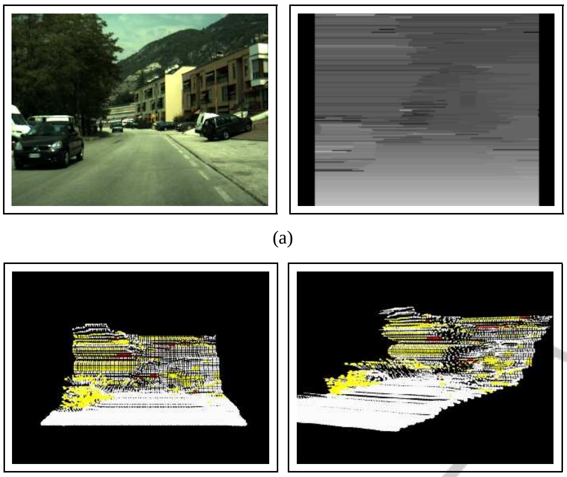


Figure 4: 3D Representation of MVFs generated from the simulated framework (Simore). (a) Left, an acquired image from the mounted stereo camera head in Simore. Right, the generated MVF. (b) Left, the spatial positioning error of direct 3D representation of disparity maps. Right, the result of the 3D MVF representation of the proposed approach.

The second data set is representing a real stereo image sequence squired from a stereo system mounted on a moving car[1]. The proposed approach has successfully modeled the 3D spatiotemporal in-

[1] Distributed Processing of Local Data for On-Line Car Services, a DIPLODOC road stereo sequence, <http://tev.fbk.eu/DATABASES/road.html>


(a)

(b)

Figure 5: 3D Representation of MVFs generated from the DIPLODOC road stereo sequence. (a) Left, an acquired image from the mounted stereo camera. Right, the generated depth map. (b) The result of the 3D MVF representation of the proposed approach.

formation from the generated depth maps as shown in fig. 5.

The proposed approach for 3D MVFs representation is very important to the 3D motion segmentation process, especially where the scene ground is heavily textured which results on generating reasonable amounts of MVs. Such MVs of the scene ground should not interfere with other MVs in the 3D motion segmentation process, otherwise false results will be generated. The accurate positioning of such MVs gives the ability to easily detect and eliminate them before starting the process of 3D motion segmentation.

Furthermore, detecting a predefined motion pattern as shown in fig. 6 has been achieved where the cone is moving to the left while the robot is slowly moving forward and the ball size is increasing, and also in fig. 5 where the mounted stereo system is moving forward. The MVs that present the translation in the z direction (which describes possible upcoming object movement) are represented in yellow. In the first data set, the MVs representing the expanding size of the ball have been detected as a possible collision, while in the second data set, the detected possible collision were the upcoming car as well as the tree behind it and some part of the background scene.

On the other hand, the proposed approach has a significant reduction of the total number of iterations required for the 3D motion segmentation process which leads to a noticeable computational time improvement. Fig. 7 shows the progression of the root mean square error $E_k(c(p_m))$ over the total iteration steps k of the previously represented synthetic MVF for the proposed algorithm compared to the segmentation approach in (Shafik and Mertsching, 2008).

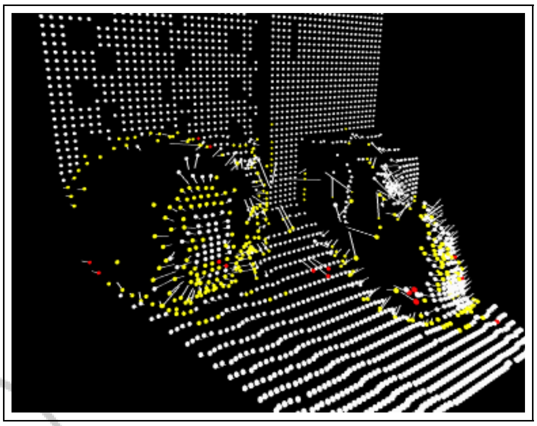


Figure 6: Detection of 3D motion patterns, yellow MVs represent the translation in the z direction which represents a possible collision with the robot.

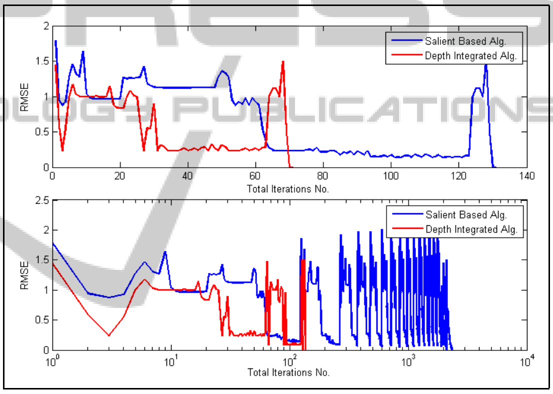


Figure 7: Progression of the root mean square error $E_k(c(p_m))$ over the total iteration steps k of the previously represented synthetic MVF for the proposed depth-integrated algorithm compared to the segmentation approach in (Shafik and Mertsching, 2008).

5 CONCLUSIONS

We have presented a fast depth-integrated 3D motion parameter estimation approach which enhanced the overall computation time of a 3D salient-based motion segmentation algorithm. In addition, the presented 3D motion parameters representation algorithm has taken into consideration the perspective transformation and the depth information to accurately position motion vectors of the generated depth sequence in the 3D space using the geometrical information of the stereo camera head. Moreover, the proposed approach has successfully detected and estimated predefined motion patterns describing important 3D motions such as movements toward the robot which is very helpful in detecting possible future collisions of moving objects with the robot.

REFERENCES

Ali, I. and Mertsching, B. (2009). Surveillance System Using a Mobile Robot Embedded in a Wireless Sensor Network . In *International Conference on Informatics in Control, Automation and Robotics*, volume 2, pages 293 – 298, Milan, Italy.

Aziz, Z. and Mertsching, B. (2009). Visual Attention in 3D Space. In *International Conference on Informatics in Control, Automation and Robotics*, volume 2, pages 471 – 474, Milan, Italy.

Bruhn, A., Weickert, J., Feddern, C., Kohlberger, T., and Schnorr, C. (2005). Variational optical flow computation in real time. 14(5):608–615.

Gruber, A. and Weiss, Y. (2007). Incorporating non-motion cues into 3d motion segmentation. *Computer Vision and Image Understanding*, 108(3):261–271.

Hasler, N., Rosenhahn, B., Thormhlen, T., Wand, M., Gall, J., and H.-P.Seidel (2009). Markerless motion capture with unsynchronized moving cameras. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'09)*.

Kim, J.-H., Chung, M. J., and Choi, B. T. (2010). Recursive estimation of motion and a scene model with a two-camera system of divergent view. *Pattern Recognition*, 43(6):2265 – 2280.

Kotthäuser, T. and Mertsching, B. (2010). Validation vision and robotic algorithms for dynamic real world environments. In *Simulation, Modeling and Programming for Autonomous Robots*, LNAI 6472, pages 97 – 108. Springer.

Li, H., Hartley, R., and Kim, J. (2008). A linear approach to motion estimation using generalized camera models. In *CVPR08*, pages 1–8.

Massad, A., Jesikiewicz, M., and Mertsching, B. (2002). Space-variant motion analysis for an active-vision system. In *Proceedings of ACIVS 2002*, Ghent, Belgium.

Pundlik, S. J. and Birchfield, S. T. (2006). Motion segmentation at any speed. In *Proceedings of the British Machine Vision Conference*, Scotland.

Ribnick, E., Atev, S., and Papanikolopoulos, N. (2009). Estimating 3d positions and velocities of projectiles from monocular views. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(5):938–944.

Rosenhahn, B., Brox, T., and Weickert, J. (2007). Three-dimensional shape knowledge for joint image segmentation and pose tracking. *Int. J. Comput. Vision*, 73(3):243–262.

Schmudderich, J., Willert, V., Eggert, J., Rebhan, S., Goerick, C., Sagerer, G., and Korner, E. (2008). Estimating object proper motion using optical flow, kinematics, and depth information. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 38(4):1139 –1151.

Shafik, M. and Mertsching, B. (2007). Enhanced motion parameters estimation for an active vision system. In *7th Open German/Russian Workshop on Pattern Recognition and Image Understanding, OGRW 07*, Ettlingen, Germany.

Shafik, M. and Mertsching, B. (2008). Fast saliency-based motion segmentation algorithm for an active vision system. In *Advanced Concepts for Intelligent Vision Systems (ACIVS 2008)*, Juan-les-Pins, France.

Shafik, M. and Mertsching, B. (2009). Real-Time Scan-Line Segment Based Stereo Vision for the Estimation of Biologically Motivated Classifier Cells. In *KI 2009: Advances in Artificial Intelligence*, volume 5803 of *LNAI*, pages 89 – 96.

Sotelo, M., Flores, R., Garca, R., Ocaa, M., Garca, M., Parra, I., Fernndez, D., Gaviln, M., and Naranjo, J. (2007). Ego-motion computing for vehicle velocity estimation. In Moreno-Daz, R., Pichler, F., and Quesada-Arencibia, A., editors, *EUROCAST*, volume 4739 of *LNCS*, pages 1119–1125.

Taylor, J., Jepson, A., and Kutulakos, K. (2010). Non-rigid structure from locally-rigid motion. In *Proceedings of Computer Vision and Pattern Recognition 2010*, San Francisco, CA.

Tsao, T.-R., Shyu, H.-J., Libert, J. M., and Chen, V. C. (1991). A lie group approach to a neural system for three-dimensional interpretation of visual motion. *IEEE Trans. on Neural Networks*, 2(1):149–155.

Yamasaki, T. and Aizawa, K. (2007). Motion segmentation and retrieval for 3-d video based on modified shape distribution. *EURASIP Journal on Advances in Signal Processing*, 2007:Article ID 59535, 11 pages.

Yang, S. and Wang, C. (2009). Multiple-model ransac for ego-motion estimation in highly dynamic environments. In *IEEE International Conference on Robotics and Automation (ICRA '09)*, pages 3531–3538.