On the Real-time Modelling of a Robotic Scene Perception and Estimation System

Sorin M. Grigorescu, *Member, IEEE,* Gigel Macesanu, Tiberiu T. Cocias and Florin Moldoveanu *Member, IEEE*

Abstract— **In this paper a real-time machine vision system for robotic scene perception and estimation is proposed. Its goal is to determine the 3D structure of the imaged scene together with the relative position and orientation of the robot with respect to the environment. Mainly, the vision system is composed of two basic elements: a 3D *camera-scene depth computation* method and a *camera egomotion estimator*. The image processing operations have been implemented into a sequential machine vision pipeline. A performance evaluation of the proposed approach is given through experimental results within an indoor environment.**

I. INTRODUCTION

One of the main components of a robotic application is represented by the robot's machine vision system. In the robotics community, it has been noticed over the past decade how the vision algorithms evolved from their classical 2D visual servoing approaches to complex 3D visualization systems that perceive both the 3D structure of the environment, as well as the relative pose (*position and orientation*) of the robot with respect to the imaged scene [1]. The computation of the robot's pose over time is determined relative to the robot's camera pose, a process also known as *egomotion estimation*.

In 3D scene perception, there are commonly two types of vision sensors used for providing visual information data: *stereo vision cameras* and *range sensors* such as laser scanners or 3D *Time-of-Flight* (ToF) cameras [2]. In the process of stereo vision based 3D perception and egomotion estimation, the stereo correspondence problem has to be solved, i.e. the corresponding feature points, necessary for 3D reconstruction, have to be extracted from both stereo images [3]. In contrast, stereo vision range sensing devices provide direct capturing of 3D scenes, delivering a pure stereo depth image in form of 3D point clouds [4]. In the case of range sensors, the obtained depth information can have different error values, depending on the sensed surface. This phenomenon makes stereo vision a more reliable solution for autonomous robotic systems that operate in real world environments.

Camera motion estimation, or egomotion, has been studied within the *Simultaneous Localization and Mapping* (SLAM) context [5]. Using detected visual information, motion estimation techniques can provide a very precise egomotion of the robot. In SLAM also, the basic sensor used is the stereo camera [6], [7]. The main operation involved in stereo based robotic perception is the computation of the so-called correspondence points used for calculating the 3D pose of the robot's camera. Correspondence calculation is also known as *feature matching*, a process which, in the case of 3D sensing, has to be applied on the stereo image pairs and also on consecutive sequence of stereo images [8]. The most common features used in this context are points localized through corner detectors such as *Harris* [9]. Another well-known technique for feature detection is the *Scale Invariant Feature Transform* (SIFT) [10], in which the computed features are invariant to rotation, scaling, translation and partially to changes in illumination. Other approaches are based on the extraction of lines or edges [7], [11] which have the advantage to provide geometrical information about the environment, but are computational expensive. Based on the extracted features, the robot's motion can be extracted with the help of estimators such as the *Kalman* [12] or *Particle Filter* [13].

In this paper the authors propose a real-time dual egomotion and scene estimation system for the purpose of reliable 3D robotic perception. The novelty of the presented research work lies on the developed 3D perception pipeline used to fuse color stereo images with the computed egomotion of the camera, as well as on its usage to sense cluttered indoor robotic environments.

The rest of the paper is organized as follows. In Section II the theory behind 3D depth sensation is presented, followed in Section III by the egomotion estimation description. The implementation of the 3D perception pipeline is given in Section IV together with performance evaluation results. Finally, conclusions and outlook are presented in Section V.

II. MECHANICS OF DEPTH SENSATION

The 3D depth estimation approach used in this paper is based on a triangulation algorithm between the left and right images of a stereo camera and an imaged real world 3D point [14]. The principle of the method is illustrated in Fig. 1.

This work was supported by the Sectoral Operational Program Human Resources Development (SOP HRD).

The authors are with the Department of Automation, Transilvania University of Braşov, Mihai Viteazu 5, 500174, Braşov, Romania `{s.grigorescu, gigel.macesanu, tiberiu.cocias, moldof}@unitbv.ro`

Source code for the proposed scene perception system can be found at `rovis.unitbv.ro`

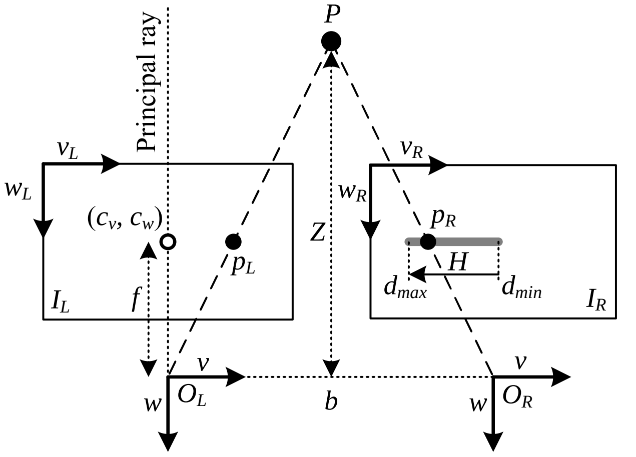

Fig. 1. 3D position estimation of a point P using a pair of rectified stereo images.

A. Stereo Camera Configuration

In order to calculate the 3D position of a point, its corresponding 2D projections onto the left and right image planes of the stereo camera have to be determined [14]. The projection can only be calculated if the stereo camera is calibrated, namely if its *intrinsic* and *extrinsic parameters* are known. These parameters are obtained through the camera calibration process which calculates on one hand the internal, or intrinsic, camera parameters K, such as the focal length f, optical centers (c_v, c_w) of both sensors, aspect ratio and skew constant and, on the other hand, the external, or extrinsic parameters, represented by the *rotation* and *translation* $[R, t]$ of each sensor of the stereo camera with respect to a fixed world reference point. Once these parameters are known, they can be used to compute the projection matrices $Q_L, Q_R \in \Re^{3\times 4}$ of each sensor, in the following homogeneous matrix multiplication form:

$$Q = \begin{bmatrix} f & 0 & c_v & 0 \\ 0 & f & c_w & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} r_{00} & r_{01} & r_{02} & t_x \\ r_{10} & r_{11} & r_{12} & t_y \\ r_{20} & r_{21} & r_{22} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = K \cdot [R|t], \tag{1}$$

where K is the intrinsic camera matrix. In the proposed system, the pose of the right sensor is always calculated with respect to the left sensor. Having in mind that the two are parallel to each other and on the same x axis, the pose of the right sensor is obtained as a pure translation with respect to the left one, namely:

$$[R_R|t_R] = [I|[b, 0, 0]^T], \tag{2}$$

where I is the identity matrix. The position of the right sensor is given by its translation over the x axis of the Cartesian space by a coefficient b known as the *baseline* between the optical centers of the two sensors.

The triangulation method can be efficiently used only if the projections of the object points on the image planes are known, that is, only if the correspondence of each feature point is known in both the left and right images. This procedure, also encountered under the name of *feature matching*, determines for each real world 3D point $P = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T$, its corresponding projections onto the left and right image planes of the stereo camera:

$$\begin{cases} p_L = \begin{bmatrix} v_L & w_L & 1 \end{bmatrix}^T, \\ p_R = \begin{bmatrix} v_R & w_R & 1 \end{bmatrix}^T, \end{cases} \tag{3}$$

where (v, w) are pixels within the 2D image plane. As can be seen from Fig. 1, the 2D image positions p_L and p_R are projected as the intersection of the image planes with the line connecting point P in world coordinates with the optical centers O_L and O_R of the stereo camera. The *principal plane*, that is the image, is located at the focal distance f from the optical centre of the sensor. The origin of the image coordinate system is considered as the top-left image corner (v_0, w_0).

B. Disparity Computation

Having in mind the calculated stereo camera parameters and points correspondences, the goal of depth computation is to determine the camera-scene distance Z. The output of the depth estimation procedure is a depth map $f(x, y, z)$ containing *voxels* representing the estimated 3D positions of the matched feature points in meters.

As already explained, the feature matching procedure determines for each pixel in the left image $I_L(v, w)$ its corresponding point in the right image $I_R(v, w)$. A straightforward approach to feature matching is to compare each pixel in $I_L(v, w)$ with each pixel $I_R(v, w)$. In this paper, a pair of pixels $\{p_L, p_R\}$ are considered corresponding points if their normalized cross correlation is the highest.

However, having in mind the high computational complexity (e.g. for a pair of typical 1024x768px images a number of app. 6^{11} calculations have to be performed), this approach is not feasible for a real-time 3D perception system. The problem has been solved through the usage of *rectified stereo images* which have the property that the pixels on the v axes of $I_L(v, w)$ and $I_R(v, w)$ are parallel to each other. Image rectification is obtained via the projection matrices Q_L, Q_R and the epipolar geometry constraint [14].

The above simplification allows the corresponding search to be performed only along the v axis of the image plane, usually on a predefined interval $H = [d_{min}, d_{max}]$, also known as the *horopter*. Hence, the 3D position estimation of a point is obtained using the so-called *disparity computation*:

$$X = (v_L - c_v) \cdot \frac{b}{d}, \tag{4}$$

$$Y = (w_L - c_w) \cdot \frac{b}{d}, \tag{5}$$

$$Z = f \cdot \frac{b}{d}, \tag{6}$$

where d is the disparity of the projected point P:

$$d = v_L - v_R. \tag{7}$$

One of the most popular disparity computation algorithm, also used in this work, is the so-called *Block Matching* (BM) [3] approach. In order to obtain a 3D model of the imaged scene, each calculated depth map $f(x, y, z)$ has to be associated with its corresponding camera pose, as will be further explained.

III. CAMERA EGOMOTION ESTIMATION

Self-localization is one of the main requirements in the field of autonomous robotics. This process, known as *egomotion estimation*, is normally accomplished by estimating the pose of the robot's stereo camera while it moves through space. In this paper, we have considered the problem of egomotion estimation through stereo vision only, as illustrated in Fig. 2.

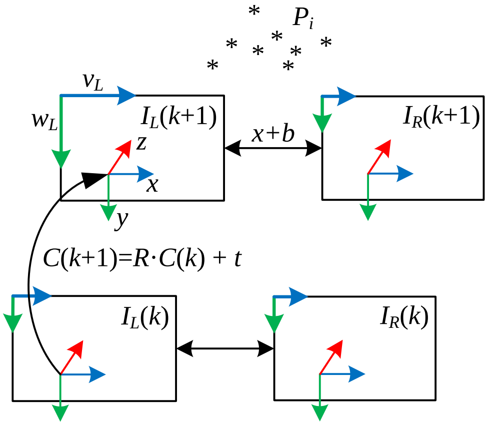

Fig. 2. Stereo vision based egomotion estimation principle.

The goal of egomotion estimation is to determine the new camera pose $C(k+1)$ with respect to the previous one $C(k)$ given a set of matched 3D points P_i and the intrinsic and extrinsic camera matrices. The P_i set of 3D points are needed to establish the *homography*, or *perspective transformation*, between the two neighboring views of the camera, that is, $C(k)$ and $C(k+1)$. A set of minimum 4 points are needed to determine the homography between two views [14]. Since only a small number of points are needed for calculating the homography, there is no need to obtain a dense, computation expensive, depth map such as the one obtained via the BM procedure explained in the previous section. On the other hand, in order to obtain a precise reconstruction of the camera's egomotion, the corresponding projected 2D points have to be precisely detected in the stereo images.

In the proposed system, a Harris corner detector [9] feature matching algorithm is used to compute from P_i the following corresponding pairs:

- a set of corresponding 2D points $\{p_{Li}(k), p_{Ri}(k)\}$ from the left and right images at camera pose $C(k)$;
- the reconstructed 3D positions $P_{RCi}(k)$ of points P_i from $\{p_{Li}(k), p_{Ri}(k)\}$ and the camera's intrinsic and extrinsic information;
- the corresponding 2D points $p_{Li}(k+1)$ in the new camera pose $C(k+1)$ for every matched 2D point $p_{Li}(k)$ in the left image of the vision sensor.

The above mentioned pairs are used in a *Perspective-N-Point* estimation manner for obtaining the new camera pose $C(k+1)$ as follows. The reconstructed 3D points $P_{RCi}(k)$, determined from the stereo pair $C(k)$ are reprojected onto the left image of $C(k+1)$. Knowing, with respect to the left image, the 2D corresponding points $p_{Li}(k) \rightarrow p_{Li}(k+1)$, a reprojection error can be calculated as:

$$E_d(p_{Li}, \widehat{p}_{RCi}) = \sqrt{(v_{Li} - \widehat{v}_{RCi})^2 + (w_{Li} - \widehat{w}_{RCi})^2}, \tag{8}$$

where $\widehat{p}_{RCi}$ are the image reprojected points from $P_{RCi}(k)$ and $E_d(\cdot)$ is the Euclidean distance. The reprojection error from Eq. 8 is further minimized through a Gauss-Newton optimization procedure, which adapts the new camera's rotation and translation $[R(k+1)|t(k+1)]$. The new pose of the camera is thus given by the values of $[R(k+1)|t(k+1)]$ which minimize Eq. 8. Finally, the new pose is calculated as:

$$C(k+1) = R(k+1) \cdot C(k) + t(k+1). \tag{9}$$

IV. REAL-TIME 3D MACHINE VISION PIPELINE AND EXPERIMENTAL RESULTS

The proposed 3D perception approach has been implemented as the image processing pipeline from Fig. 3. The first operation on the pipeline is the image acquisition one, which reads color images from the stereo camera. Once the left and right image pair is available, two distinct processing threads are started. The first thread computes the depth image, which contains dense camera-object distances, while the second thread deals with feature matching and the calculation of the camera's egomotion. When both the camera's pose $[R|t]$ and the depth map $f(x, y, z)$ are available, they are annotated to the virtual 3D model of the imaged scene. The 3D coordinates of the depth map are rendered with respect to the computed camera pose.

We have considered a processing cycle to begin with image acquisition and end with camera and depth annotation to the virtual 3D model, as depicted in Fig. 3. In the experimental setup, a mechanically calibrated Point Grey Bumblebee® stereo camera system has been used to acquire a sequence of 519 indoor images. The average computational time needed by the considered processing operations is shown on the timeline from Fig. 3. As can be seen, a processing cycle lasts just over 700ms, the value being low enough to consider the proposed system as a real-time one. The implemented image processing program has been tested on a typical portable computer running a 64 bits operating system on an Intel® i3 dual core CPU, each processor having a 2.40GHz clock speed.

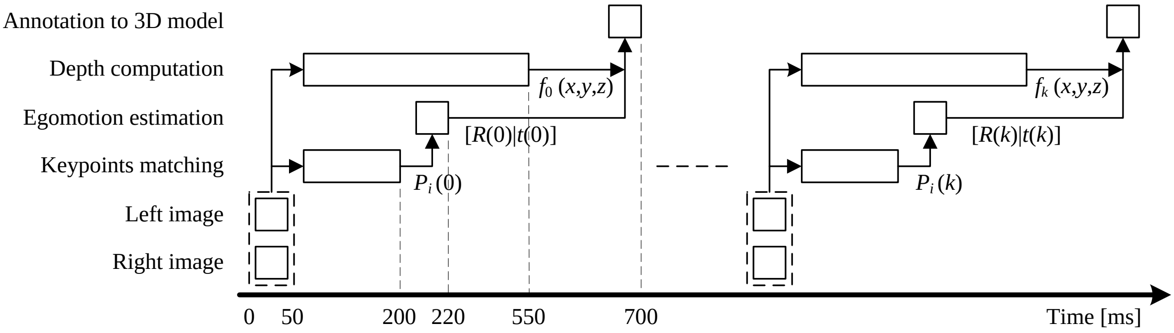

Fig. 3. Computational pipeline of the proposed 3D perception system.

An snapshot of a processing slice can be seen in Fig. 4. One of the main operations in egomotion estimation is the calculation of the reprojection error, shown in Fig. 4(a). The reprojection error is represented as the blue line linking the correspondences between the $C(k)$ and $C(k+1)$ left images (yellow circles) and the 3D reprojected points from $C(k)$ onto the $C(k+1)$ left image (red circles). An example of egomotion estimation and depth map annotation over a sequence of 15 images is presented in Fig. 4(b). The rendering of the annotated 3D virtual model is performed on top of an OpenGL® render.

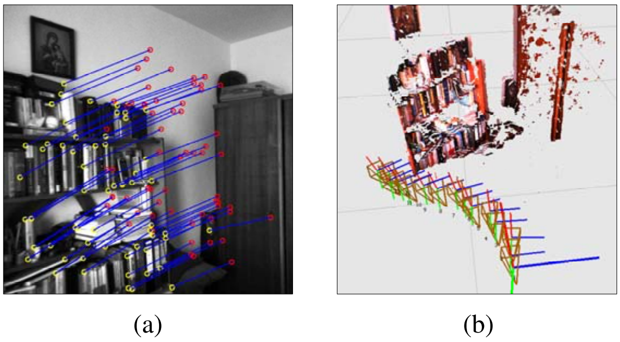

Fig. 4. Slices taken from the 3D perception pipeline (best viewed in color). (a) The reprojection error (8) over a typical indoor scene. (b) Camera egomotion and scene structure annotation over a sequence of 15 stereo images.

V. CONCLUSIONS AND OUTLOOK

In the presented paper, a real-time 3D perception system has been proposed. Its main elements are the depth sensing algorithm together with the camera egomotion estimator. The goal of the proposed system is to be used on autonomous robotics platforms that need a real-time 3D machine vision component. The system has been implemented as a real-time processing pipeline, one full operational cycle being computed in just over 700ms, thus meeting the real-time requirements.

As future work, the authors consider the speed enhancement of the proposed system using state of the art parallel processing equipment, such as FPGAs and GPUs which, in the authors opinion, would exponentially decrease the processing time (e.g. feature matching for the left and right images, as well as matching for neighboring camera poses would be done in parallel). Also, a real-time feedback optimization technique of the proposed perception system is considered for coping with uncertain situations that can occur in the imaged environment (e.q. variable illumination, moving objects, occluded areas etc.).

ACKNOWLEDGMENT

This paper is supported by the Sectoral Operational Program Human Resources Development (SOP HRD), financed from the European Social Fund and by the Romanian Government under the projects POSDRU/89/1.5/S/59323, POSDRU/88/1.5/S/59321 and POSDRU/107/1.5/S/76945.

REFERENCES

[1] A. Andreopoulos, S. Hasler, H. Wersing, H. Janssen, J. Tsotsos, and E. Koerner, "Active 3D Object Localization Using A Humanoid Robot," *IEEE Trans. on Robotics*, vol. 27, no. 1, pp. 47–64, 2011.

[2] S. Hussmann and T. Liepert, "Robot Vision System based on a 3D-TOF Camera," in *Instrumentation and Measurement Technology Conference-IMTC 2007*, (Warsaw, Poland), 2007.

[3] M. Brown, D. Burschka, and G. Hager, "Advances in Computational Stereo," *IEEE Trans. on Pattern Recognition and Machine Intelligence*, vol. 25, no. 8, pp. 993–1008, 2003.

[4] R. Rusu, Z. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D Point Cloud based Object Maps for Household Environments," *Robotics and Autonomous Systems*, vol. 56, pp. 927–941, 2008.

[5] J. Neira, A. Davison, and J. Leonard, "Special Issue on Visual SLAM," *IEEE Trans. on Robotics*, vol. 24, no. 5, 2008.

[6] L. Paz, P. Pinies, J. Tardos, and J. Neira, "Large-Scale 6-DOF SLAM With Stereo-in-Hand," *IEEE Trans. on Robotics*, vol. 24, no. 5, pp. 946–957, 2008.

[7] T. Lemaire, C. Berger, I. Jung, and S. Lacroix, "Vision-Based SLAM: Stereo and Monocular Approaches," *Int. Journal of Computer Vision*, vol. 74, no. 3, pp. 343–364, 2007.

[8] A. Geiger, J. Ziegler, and C. Stiller, "StereoScan: Dense 3D Reconstruction in Real-time," in *IEEE Intelligent Vehicles Symposium*, (Baden-Baden, Germany), June 2011.

[9] C. Harris and M. Stephens, "A Combined Corner and Edge Detection," in *Proc. of the Fourth Alvey Vision Conference*, pp. 147–151, 1988.

[10] D. G. Lowe, "Object Recognition from Local Scale-invariant Features," in *Int. Conf. on Computer Vision*, (Corfu, Greece), pp. 1150–1157, Sept. 1999.

[11] E. Eade and T. Drummond, "Edge Landmarks in Monocular SLAM," in *British Machine Vision Conference-BMVC 2006*, (Edinburgh, U.K.), pp. 469–476, 2006.

[12] G. Welch and G. Bishop, *An Introduction to the Kalman Filter*. UNC-Chapel Hill, 2006.

[13] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Trans. on Signal Processing*, vol. 50, no. 2, 2002.

[14] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.