Chapter 11
Vision-Based Navigation Strategies

Darius Burschka

11.1 Motivation

Navigation and localization are important capabilities of mobile systems allowing definition of mission goals. Only the knowledge about the absolute and relative position in an indoor or outdoor environment allows a free definition of mission goals and path planning in areas not previously traversed by the system. A typical concurrent goal is the reconstruction of coherent 3D geometric representations of arbitrary indoor or outdoor environments from a configurable set of sensors. This representation is typically used as a reference for localization. The sensor configuration is thereby defined by the required accuracy and system costs. We investigate monocular and binocular cameras, laser range finders, and inertial systems as input sources for this task. The minimal hardware configuration of such a system is a monocular camera that can be supported by additional sensors to enhance the quality of the reconstructed models. The goal is to replace expensive inertial systems with a set of low-cost sensors, like video cameras available on most current computer systems. The necessary accuracy is achieved through fusion of information over a sequence of images. The idea is to replace expensive hardware with appropriate algorithmic techniques to compensate for the imperfections of the low-cost sensors.

An important milestone towards a high accuracy reconstruction of the environment is an exact localization in an unknown or partially known environment. The reference model for localization needs often to be extracted in parallel to the actual localization task. This process is known in the literature as *Simultaneous Localization and Mapping* (SLAM). The localization is necessary to fuse the sensor readings from different positions to a consistent and complete 3D model.

In this chapter, we will focus on the localization task from a video camera. We assume a video camera mounted on a mobile system. The localization implicates several challenges. The first challenge is an accurate estimation of the 3D pose pa-

Technische Universität München, Department of Computer Science, 80333 München, Germany, e-mail: burschka@cs.tum.edu

rameters from the available sensor data. Another challenge is to perform the localization in situations, where the reference points or landmarks as we will refer to them in the following text are not known a-priori and need to be estimated in parallel to the localization process. We propose systems that are capable of simultaneous localization of the camera and navigation relative to obstacles in the world.

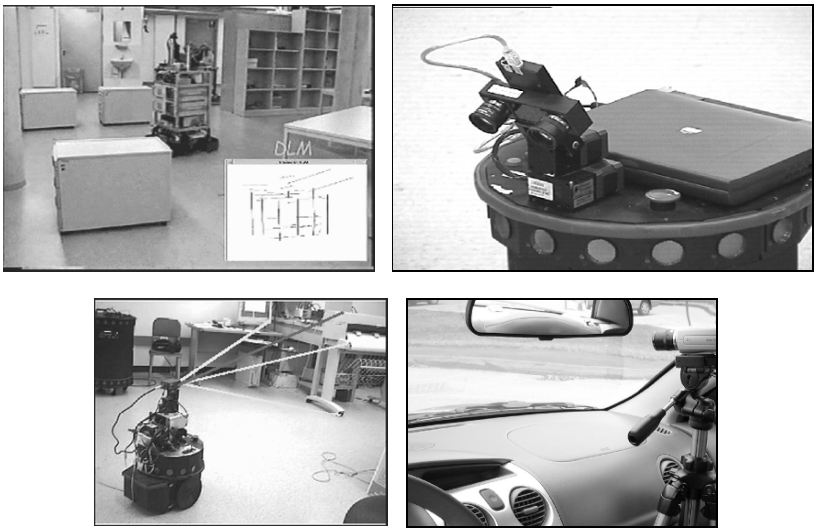

Fig. 11.1 Different types of navigation systems using video cameras: (Marvin) 3D exploration of indoor environments with stereo [4], (Speedy) obstacle avoidance from dense disparity maps [7], (Goomba) sentry robot using vision-based control for navigation [5], (car application) Visual SLAM for traffic sign detection [8].

The problem of *Simultaneous Localization and Mapping*, also known as SLAM, has attracted immense attraction especially in the mobile robotics literature. SLAM addresses the problem of building a map of an environment from a sequence of landmark measurements obtained from a moving system. Since the motion especially of hand-operated devices is unknown, the mapping problem induces a localization problem. The partial 3D reconstructions can only be fused to a complete model given an accurate relative localization between them. A solution to the SLAM problem using Kalman Filters was introduced in a paper by Smith, Self, and Cheeseman [23]. This paper proposed the use of the Extended Kalman Filter (EKF) for incremental estimation of the posterior distribution over the robot pose along with the positions of the landmarks. While many popular SLAM implementations use laser range information as input to the process to simplify the estimation to pure localization task, we present an extension to vision-based techniques. The challenge here is to obtain the necessary information for the SLAM process from a monocular camera as input source.

We developed a variety of navigation systems using different approaches ranging from mission planning based on explored 3D models from a binocular setup [4], over localization relative to obstacles in the world from stereo [7] to monocular approaches based on visual servoing [5] and a recently developed visual *simultaneous localization and mapping* system ((VGPS)SLAM) [10] (see Fig. 11.1). We will give a short evaluation of the advantages and disadvantages of these systems and discuss the next steps in our current research.

11.1.1 Related Work

The problem that we address here is a simultaneous estimation of the motion parameters R,T (rotation and translation) and the depth information as a metric distance to the observed points. We address the extension of typical SLAM based on laser range finders to monocular cameras.

There exist solutions to pose estimation for 3 point correspondences for most traditional camera models, such as for example orthographic, weak perspective [1], affine, projective [14, 18] and calibrated perspective [19]. These approaches constrain the possible poses of the camera to up to four pairs of solutions in the case of a calibrated perspective camera. At most one solution from each pair is valid according to the orientation constraints and the other solution is the reflection of the camera center across the plane of the three points.

Many localization approaches for indoor applications use simplifications like assumptions about planarity of the imaged objects in the scene or assume a restricted motion in the ground plane of the floor that allows to derive the metric navigation parameters from differences in the images using Image Jacobians in vision-based control approaches. A true 6DoF localization requires a significant computational effort to calculate the parameters while solving an octic polynomial equation [22] or estimating the pose with a Bayesian minimization approach utilizing intersections of uncertainty ellipsoids to find the true position of the imaged points from a longer sequence of images [12]. While the first solution still requires a sampling to find the true solution of the equation due to the high complexity of the problem, the second one can calculate the result only after a motion sequence with strongly varying direction of motion of the camera that helps to reduce the uncertainty about the position of the physical point. In the work of Nister [22], an approach sampling for the correct solution along the rays of projection solving an octic polynomial to find the actual camera pose is presented. It is limited to exactly 3 points neglecting any possible additional information. While it represents a direct solution to the problem, the high order of the polynomial and the typical noise in real images makes this solution still very complicated and sensitive to noise.

Our system is motivated by the same idea as the system presented in [17], where a tracking approach for "2.5D space" was proposed. The system is supposed to compensate for the drawbacks of classical position-based visual servoing. In the approach presented in [17], eight landmarks are necessary to estimate the pose of an

object in space. A reduction to four points is only possible in case that four co-planar points can be identified. The co-planarity constraint is a special case that is difficult to enforce in all situations. Additionally, a robust tracking of eight landmarks in the image is contradictory to our goal to build a compact system running on hardware with limited computational power that can usually be found on mobile systems. The smaller the number of landmarks that we need to track, the more processing power can be dedicated to other important tasks on the robot.

Our pose estimation is based on an image-based approach that compares the 2D projections of an *internal 3D model* between images. The *internal 3D model* is estimated up to scale due to the limitations in the perception of a monocular camera system (see Section 11.3.3.1). In [11] a recursive model-based object pose estimation is presented that is based on orthographic projection of points onto camera image. This approach is limited to configurations that can be projected onto a planar image. In our case, we propose a pose estimation method allowing robust pose verification from 3 tracked landmarks that can be placed anywhere around the sensor. Our approach operates in image coordinates of the camera using a novel representation for the 3D model that does not require any knowledge about the three-dimensional position in the world to register the reconstructions to each other.

We propose an approach that we validated in a wide range of applications ranging from reconstructions from endoscopic medical images to 3D scene reconstructions in outdoor environments.

We assume to know the initial 3D structure of at least 3 points in the world P_i with known correspondences in the image frame n_i. The system is initialized manually or automatically with an initial set of feature correspondences with a known metric relation and it maintains these correspondences through tracking in color or texture. It adds new features to the set to compensate for loss of features that become occluded or that disappear from the field of view. Further, we assume also a calibrated camera measuring directly the angles of incidence. In this chapter, we focus on strategies for depth recovery from spherical projection and the motion+structure update.

In the following Section 11.2, we give an overview of possible navigation approaches with a discussion of their advantages and disadvantages. In Section 11.3, we describe the way the information about the depth changes due to motion and how the motion itself is calculated. In Section 11.3.3, we discuss the open challenges for our monocular navigation system. The accuracy of the algorithm is evaluated in Section 11.4. We conclude in Section 11.5 with an overall evaluation of the presented system and present our future research goals.

11.2 Navigation Alternatives

In this section, we present an overview of navigation approaches that we implemented on our mobile systems in the course of the past years. They lead us to our fi-

nal approach based on monocular VSLAM. We also discuss lessons that we learned about the advantages and disadvantages of each of the approaches.

We distinguish between map-based systems that build an intern representation of the environment as a 2D- or 3D-model and image based approaches deriving the information directly from an error between an expected and an observed position of a physical point in the world.

11.2.1 Map-based Navigation

Map-based navigation systems represent an approach to global navigation. The navigation is based on 2.5D or 3D models of the environment. A 3D model as a global reference allows a localization relative to a specific physical reference point in the world in opposite to a relative localization between two sensor frames. Relative localization is common in image-based approaches.

The sensor information is abstracted to a 3D representation and fused from all sensor readings to a consistent global or local model of the environment. The model allows planning of arbitrary missions in the environment that may traverse locations which were only perceived by the sensors, but which were never actually passed in previous missions. It is possible, because the 3D model allows a prediction of any new sensor view in the world even for new locations. Their advantage is the flexibility allowing planning of arbitrary missions even in regions which were not traversed before, but a significant disadvantage is the necessity of fusion of information from the sensor readings requiring an exact localization over a long period of time to allow a correct registration of all sensor readings in an area. An additional disadvantage is the abstraction of the information from the direct sensor data to three-dimensional descriptions which are prone to errors due to calibration errors in the system.

11.2.1.1 Binocular 3D Reconstruction

At the Lab for Real-Time Computer Systems of the Technical University in Munich, we built a mobile robot *Marvin* that reconstructs the 3D world model from the perception of a binocular camera system [4]. The system is depicted in Fig. 11.2.

The system reconstructs three-dimensional line segments representing the boundaries of human made objects. The line segments are stored in a local map (DLM) fusing the consecutive sensor readings from the sensor in Fig. 11.3. The map is the central element of the navigation system. It decouples the three major information flow loops in the system marked as colored regions in Fig. 11.3. They all operate with different cycle times. There is a fast bidirectional information exchange between the local map DLM and the sensor system that predicts expected information for the current view based on the 3D map content and stores back the current reconstruction.

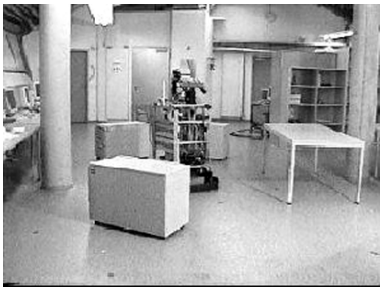

Fig. 11.2 Exploration system *Marvin* using binocular stereo.

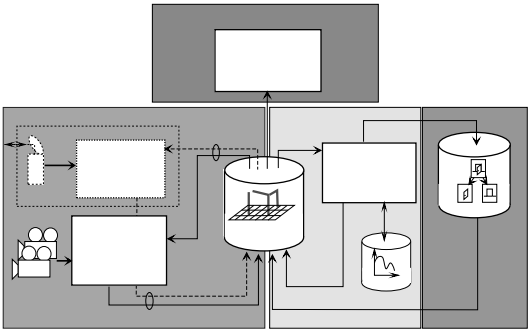

Fig. 11.3 The information flow in the map-based exploration system *Marvin*.

This loop helps to reduce false matches in stereo processing (3D reconstruction module), because expected information is predicted for each step increasing the matching score of correct matches between line segments in the stereo images. It helps also to filter out wrong matches that cannot be verified in a current frame in Fig. 11.4.

This filtering is based on the assumption that correct matches will always occur close to their true position with a small positional error due to localization and camera calibration errors. False matches move to varying locations depending on the viewing position. Correct entries in the map correspond to line segments that could be reconstructed from different viewing positions in a local environment.

The reconstructed line segments can be abstracted to polygons or even objects in the spatial prediction module in the right block in Fig. 11.3. This can provide additional hypotheses about missing lines based on assumptions about underlying structures that can be provided to the sensor system as local predictions to be verified or discarded in the current view. This module operates outside of the fast sensor loop and does not interfere with the sensor processing directly. The calculated information about missing parts of hypothetical objects identified in the data is inserted asynchronously to be verified in the sensor loop when the corresponding region comes into sensor view.

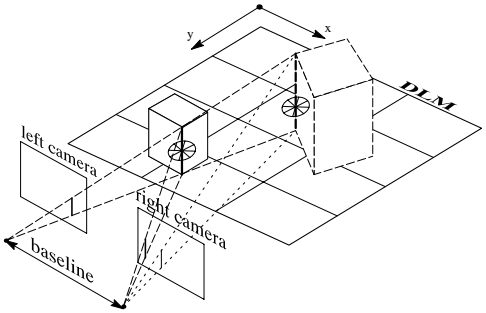

Fig. 11.4 Multiple matching candidates for a given segment can be stored in the local map DLR to be verified from a different location.

11.2.1.2 Monocular VSLAM Systems

In many cases, 3D reconstruction is necessary in large distances to the camera system to allow pre-selection of interesting objects for a mission far ahead before the system moves closer to them. A typical example is a car navigation system, where the high speed of motion requires analysis of objects in large distance to the car to give enough time for decision. An example can be a traffic sign detection system

Fig. 11.5 Large baseline can easily be constructed utilizing the motion of the vehicle with monocular systems.

that analyzes candidates for signs as soon as they become visible [10]. The relationship between the depth, z, of a scene point and its disparity, D, in two images separated by baseline B is given by [14]:

$$D = \frac{B \cdot f}{p} \cdot \frac{1}{z}, \tag{11.1}$$

where f is the focal length of the camera and p is the pixel-size on the camera chip. This relationship is shown in Fig. 11.6 for several values of the distance between the stereo cameras B. From the graph, it is clear that the larger the value D for a given landmark, the better the signal to noise ratio of the resulting reconstruction.

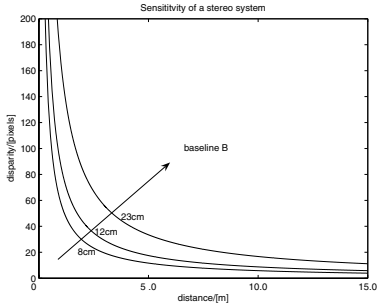

Fig. 11.6 The disparity value drops rapidly with the distance to the imaged landmark.

Some ways of increasing depth accuracy include increasing f (at the cost of field of view) or decreasing p (i.e. using a higher resolution camera). The former is generally limited by the need for a reasonably wide field of view; the latter is limited by the bandwidth and processing necessary to handle higher resolution images.

As a result, the only real flexibility is in the baseline. There are natural limits set on the maximum width of a binocular system. These limits can be defined, e.g., by the width of the car. Any further increase requires a change to a monocular reconstruction that uses the motion of the system as a baseline for reconstruction (see Fig. 11.7).

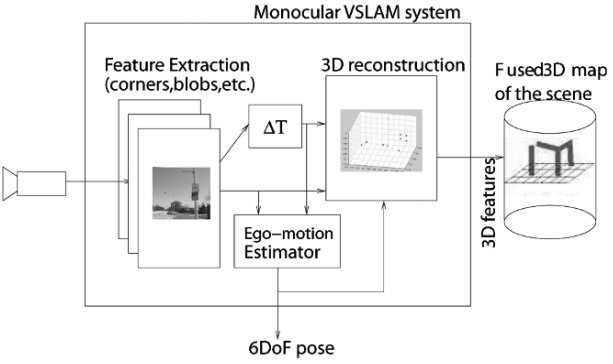

Fig. 11.7 Monocular VSLAM system tracking positions of point feature in a sequence of images to reconstruct their 3D position in parallel to estimation of the motion parameters of the camera system.

Feature points representing a specific 3D point in the world are tracked in a sequence of images to estimate both their 3D position and the relative motion of the camera to them in a SLAM *(simultaneous localization and mapping)* approach.

Like in the case of a binocular reconstruction, the resulting system constructs a global or local 3D model containing in our case points representing centers of unique patterns in the world. An accurate localization is necessary to fuse the single

reconstructions from consecutive steps making the processing more complex and sensitive to reconstruction and localization errors. In case that the system is used for absolute localization, special care needs to be taken to keep the resulting errors small. Relaxation techniques known from laser based SLAM approaches are applied to reconstructed data to minimize the error.

The monocular approach can also be used as a relative localization system providing just position changes between consecutive sensor readings. In this case, just the image position of corresponding feature points is analyzed in two image frames allowing an estimation of relative motion without a necessity of fusion with information from previous steps. Here, no global localization is performed. Our monocular SLAM implementation is a generalization of the Vision-Based Control (VBC) navigation described in Section 11.2.2.1 below. This generalization allows large displacements between the acquisition points of both images, because an analytic pose estimation is used instead of a local linearization used in the image Jacobian from VBC. In both cases, a displacement to a reference pose is calculated.

11.2.2 Image-Based Navigation

Many of the complicated house keeping methods to ensure correct global pose and exact 3D reconstruction that are necessary for correct data fusion can be avoided in image-based navigation approaches. Usually, these approaches do not provide an absolute localization relative to a physical reference point in the world. They calculate merely a relative motion between sensor readings instead. A fusion to an absolute pose can be done outside of the navigation module. This navigation method corresponds more to an inertial unit estimating just changes instead of integrating them to an absolute value. Possible localization errors appear as noise on the top of the relative pose estimation values.

11.2.2.1 Vision-Based Control

Many applications of mobile systems involve repeating tasks that require a robot to move along a pre-defined path. The system does not require significant flexibility in the choice of the paths, but high robustness is required for a long term operation. A typical task for this type of systems is a sentry robot or mail-delivery robot repeating the same paths in each mission. To avoid the localization problems of the map-based approaches that suffer from calibration errors which may occur due to vibrations during operation, these systems use directly the images as a ”model” to store the correct path. Changes in the projection between an expected and the actual position of a landmark feature are used to calculate the pose error. This is a relative localization error relative to the pose from where the reference image was taken. The system does not need to have any knowledge about the absolute pose in the world at any time.

We explain the method at an example of a robot moving in a plane of the floor that restricts its motion to the two dimensions of the plane (x,z) and the orientation Θ in Fig. 11.8.

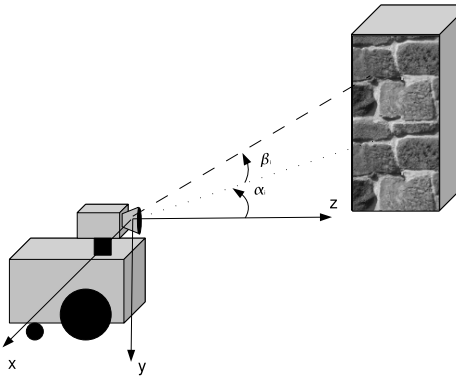

Fig. 11.8 The system estimates motion errors based on the difference in the observation between the expected and the actual position of an imaged point.

A camera system is an angle measuring device. It estimates the direction (α_i, β_i) from which a specific point in the world can be seen. The radial distance along the line of sight is lost in the projection. Each observation with the metric pixel coordinates (u_i, v_i) for a focal length f=1 can be converted into two angles (α_i, β_i) describing the azimuth and elevation values for a given observation to:

$$\alpha_i = \arctan u_i = \arctan\left(\frac{x_i}{z_i}\right), \quad \beta_i = \arctan\frac{v_i}{\sqrt{1+u_i^2}} = \arctan\frac{y_i}{\sqrt{x_i^2+z_i^2}}$$

Assuming motion in the plane, we can compute the following image Jacobian relating the change of angles in the observation, (α_i, β_i), due to changes in motion in the plane, (x_i, z_i) to:

$$\mathscr{J}^t = \begin{pmatrix} \frac{\partial\alpha_i}{\partial x_i} & \frac{\partial\alpha_i}{\partial z_i} & \frac{\partial\alpha_i}{\partial\Theta_i} \\ \frac{\partial\beta_i}{\partial x_i} & \frac{\partial\beta_i}{\partial z_i} & \frac{\partial\beta_i}{\partial\Theta_i} \end{pmatrix} = \begin{pmatrix} \frac{z_i}{x_i^2+z_i^2} & -\frac{x_i}{x_i^2+z_i^2} & -1 \\ -\frac{x_i y_i}{(x_i^2+y_i^2+z_i^2)\cdot\sqrt{x_i^2+z_i^2}} & -\frac{y_i z_i}{(x_i^2+y_i^2+z_i^2)\cdot\sqrt{x_i^2+z_i^2}} & 0 \end{pmatrix} \tag{11.2}$$

The dependency on the Cartesian coordinates can be avoided considering the geometry of the system to:

$$R_i = \sqrt{x_i^2 + y_i^2 + z_i^2} = \frac{y_i}{\sin\beta_i} \quad r_i = \sqrt{x_i^2 + z_i^2} = \frac{y_i}{\tan\beta_i}$$

$$\mathscr{J}_i^t = \begin{pmatrix} \frac{z_i}{r_i^2} & -\frac{x_i}{r_i^2} & -1 \\ -\frac{x_i y_i}{R_i^2 \cdot r_i} & -\frac{y_i z_i}{R_i^2 \cdot r_i} & 0 \end{pmatrix} = \begin{pmatrix} \frac{\tan\beta_i \cdot \cos\varphi_i}{y_i} & -\frac{\tan\beta_i \cdot \sin\varphi_i}{y_i} & -1 \\ -\frac{\sin^2\beta_i \cdot \sin\varphi_i}{y_i} & -\frac{\sin^2\beta_i \cdot \cos\varphi_i}{y_i} & 0 \end{pmatrix}$$

Note that the image Jacobian is a function of only one unobserved parameter, y_i, the height of the observed point. Furthermore, this value is *constant* for motion in the plane. Thus, instead of estimating a time-changing quantity as is the case in most vision-based control, we only need to solve a simpler static estimation problem for a constant value y_i in case of the motion in the floor plane.

This system is very robust to errors in the calibration, since the goal of the processing is to correct an image error to zero, which is independent of the estimates of the focal length and radial lens distortions. These errors usually just cause the system to assume a too large deviation. The correct alignment is still detected correctly.

11.2.2.2 Disparity-based Navigation

Obstacle avoidance systems are essential to protect robots from collisions with the environment or driving towards staircases or gaps (*negative obstacles*) while operating in unknown or partially known environments. Many obstacle avoidance systems are based on sensors that provide direct 3D measurements, such as laser range finders and sonar systems [3, 13]. In some cases, e.g. [16], cues from a monocular camera combined with prior knowledge of supporting surface geometry and appearance have been used. In contrast, our system relies completely on the data from a real-time stereo system with relative few prior assumptions.

Disparity images are pseudo-images, where each pixel value corresponds to the disparity D (reciprocal value to the depth distance z, see (11.1)). Two example of such images are depicted in Fig. 11.9 below. The goal of the binocular system is to recover all planar structures with a given size and position in space in the current camera view. In previous work [6], we describe a system that was able to recover supporting planes from binocular stereo images to detect obstacles in the scene. This approach relied on the fact that there is a homography between the (u,v,D) coordinates of a disparity image ([u,v]-image coordinates and disparity D) and the corresponding Cartesian coordinates from the 3D scene. Here we sketch how we use the idea to locate and estimate planar structures.

Following the derivation in [6], given a plane $\mathscr{P}_r$ in R^3,

$$\mathscr{P}_r : a_r x + b_r y + c_r z = d_r \tag{11.3}$$

the equivalent disparity plane is given by

$$\forall z \neq 0: \quad a_r \frac{x}{z} + b_r \frac{y}{z} + c_r = \frac{d_r}{z} \tag{11.4}$$

$$a_r u + b_r v + c_r = k \cdot D(u,v) \tag{11.5}$$

$$\text{with } u = \frac{x}{z},\ v = \frac{y}{z},\ k = \frac{d_r}{B}. \tag{11.6}$$

where $D(u,v)$ represents the disparity at image coordinates (u,v). Clearly, (11.4) describes a plane in UVD space. We can write (11.4) in the following form

$$D(u,v) = \begin{pmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \end{pmatrix} \cdot \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \mathbf{n}_\mathbf{r}^* \cdot \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \tag{11.7}$$

$$\text{with} \quad \rho_1 = \frac{a_r}{k},\ \rho_2 = \frac{b_r}{k},\ \rho_3 = \frac{c_r}{k} \tag{11.8}$$

All pixels in the disparity image that have the in (11.7) predicted disparity value are removed form the image and the remaining pixels are treated as obstacles. Ground suppression is fundamental for the entire process. An example of a suppression is shown in Fig. 11.9. It shows the resolution of the system, which is capable of distinguishing between the ground plane and objects as low as 1cm above the ground at a distance of up to 3m. The newspaper disappears as an obstacle as soon as it lays flat on the ground. Each image triple shows the real image in the upper left corner, the computed disparity image in the upper right corner and the detected obstacles at the bottom.

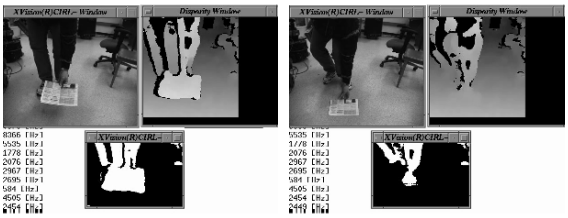

Fig. 11.9 The newspaper is classified as obstacle left, but it disappears in the right image.

The common feature of this navigation category is that all the necessary information is derived from the current sensor reading without any necessity of fusion between different readings. In this case, like already in section 11.2.2.1, we get the navigation information directly from the image itself.

11.3 Monocular VSLAM Approach

Since a typical video camera measures only the angle of incidence of the incoming rays of light, it is useful to remove the dependency on the physical imaging properties of the sensor and introduce a more generic sensor model. We decided to use a spherical projection model for our system, where every 3D-point P_i is represented as a unit vector n_i pointing in its direction:

$$n_i = \frac{P_i}{||P_i||} \quad \vee \quad n_i = \frac{(u\ v\ 1)^T}{||(u\ v\ 1)^T||} \tag{11.9}$$

We see in (11.9) that there is a simple relation between the uni-focal (focal length=1) image coordinates (u, v) and the projection on the sphere n_i.

In the remainder of this section we describe the way, how the motion parameters (R, T) and the changing 3D structure elements $\{D_i\}$ are recovered. The motion parameters represent a delta motion to the previous or a reference frame, but, for simplicity, we will omit the Δ expression in front of them. We use image-based tracking to maintain the correspondences between the image frames.

11.3.1 3D-Reconstruction

Analogous to the typical binocular approach, the 3D information is extracted using additional information from a second image or an initial 3D reference model.

11.3.1.1 Reconstruction of Unknown Points.

This processing step is necessary to recover the depth structure for new points that appeared in the camera images and need to be added to the tracking process. This processing can also be used for dual camera systems (e.g., two omnidirectional cameras) with known, calibrated displacement (R, T).

In a parallel binocular system with a distance B between the cameras, the normal distance Z to an imaged point P_i is estimated from a horizontal shift (metric disparity) d between both images [25] to

$$Z = \frac{B \cdot f}{d}, \quad \text{f - focal length of the cameras} \tag{11.10}$$

Since we deal here with a monocular system that reconstructs only sparse information about a few corresponding points, we want to avoid any warping operation to the parallel case. In typical structure-from-motion applications, the translation T is known only up to scale $\frac{T}{m}$ [25]. Therefore, we modified (11.14) in the following way:

$$\frac{D_i'}{m} n_i' = R \frac{D_i}{m} n_i + \frac{T}{m}$$

$$\begin{pmatrix} n_i' & -R \cdot n_i \end{pmatrix}^{-1} \cdot \frac{T}{m} = \begin{pmatrix} \frac{D_i'}{m} \\ \frac{D_i}{m} \end{pmatrix} \tag{11.11}$$

This is the *spherical disparity* equation with a similar structure to (11.10). The baseline B of the system is the distance $\frac{T}{m}$ traveled by the camera and it is "divided" by the *spherical disparity* s

$$s = \begin{pmatrix} n_i' & -R \cdot n_i \end{pmatrix}, \tag{11.12}$$

which represents a difference vector between the two projections (n_i', n_i) rotated to the coordinate frame of n_i' in which $\frac{T}{m}$ is defined. Since there is no significant plane as it is the case for the image plane of a coplanar binocular system, a normal distance definition of Z does not make any sense and it is replaced by the radial distance to the focal points of both projections $(\frac{D_i}{m}, \frac{D_i'}{m})$. The reconstructed depths are scaled down to the same scale as T. The scale m is preserved in the reconstruction. In the following text, we will assume m=1 to simplify the notation. It is easy to verify that all equation are true for any value of $m > 0$.

11.3.1.2 Update of Known Radial Distances.

The presented system maintains a set of known correspondences that was used to recover the motion. For these points, the depths $\{D_i\}$ in the previous frame or in the reference position at the origin are assumed to be known. The task is to update them to the current depth $\{D_i'\}$. Theoretically, the equation (11.11) can be used for this task, but since some of the data may represent very accurate model information, a different type of update equation is used. It takes the accurate depth information instead of the calculated motion information (R,T) into account.

Since we try to estimate the 6DoF motion, point and line features do not provide sufficient information to describe all 6 motion parameters. We have chosen a plane $\mathscr{E}$ spawned by 3 feature points $\{P_1, P_2, P_3\}$ and $\{P_1', P_2', P_3'\}$ in both images as a reference feature that is observed in both images of a sequence. The features must not be collinear. We construct two vectors $v_1 = P_2' - P_1' \wedge v_2 = P_3' - P_1'$ and describe the plane segment with the diagonal resulting from addition of these two vectors.

$$v_1 = P_2' - P_1' \wedge v_2 = P_3' - P_1'$$

$$\mathscr{D} = (D_1, D_2, D_3)^T$$

$$v_1 = D_2' n_2' - D_1' n_1' = R \cdot D_2 n_2 + T - R \cdot D_1 n_1 - T$$

$$v_2 = D_3' n_3' - D_1' n_1' = R \cdot D_3 n_3 - R \cdot D_1 n_1$$

$$v_1 + v_2 =$$

$$= (-2n_1' \, n_2' \, n_3') \cdot \mathscr{D}' = R \cdot (-2n_1 \, n_2 \, n_3) \cdot \mathscr{D}$$

$$F' \cdot \mathscr{D}' = R \cdot F \cdot \mathscr{D}$$

$$\Rightarrow \mathscr{D}' = F'^{-1} R \cdot F \cdot \mathscr{D} \tag{11.13}$$

The equation (11.13) introduces the projection matrix F that projects the *depth vector* $\mathscr{D}$ onto the diagonal vector $v_1 + v_2$ in the plane $\mathscr{E}$. It allows a recovery of the updated depth values $\mathscr{D}'$ based on the current image data that was used to construct F and F', and the known geometric structure $\mathscr{D}$ from the previous frame. The equation (11.13) shows that from a known set of relative distances $\mathscr{D}$ the new 3D structure D' after the motion can be reconstructed without any knowledge about the translation in the system T. It is an important property of this estimation system, since monocular systems are able to recover the rotation matrix R correctly, while the translation vector T is estimated only up to an unknown scale factor if there is no external metric reference in the world used.

11.3.2 Motion Recovery

The reconstruction approaches in the previous section (Section 11.3.1) assumed a knowledge of the motion parameters (R,T). In case of a monocular system, these parameters are unknown and need to be estimated in parallel to the reconstruction process. We mentioned already in the motivation section that there is no linear relation between the motion and structure parameters according to (11.14). The typical structure-from-motion approaches are able to reconstruct the motion from 5-8 point correspondences between the images. In our case, we assume to have additional information about the metric distances $\mathscr{D}$ to the imaged points that will allow us to reduce this number to a minimal set of 3 features. Our goal is to develop an algorithm that on one hand works with a minimum feature set but on the other hand accepts additional features if they are available. This is one of the important differences to the algorithm presented in [22] that operates on 3 points assuming their accurate detection. In real applications, the feature detection is error-prone and some of the

errors can be compensated by using additional features in over-determined systems of equations.

11.3.2.1 Motion-Induced Changes in the Feature Projections

The equation (11.11) describes completely the change in the projection $n_i \rightarrow n_i'$ due to arbitrary motion in all 6 degrees of freedom (R,T). The motion estimation needs to be separated from the reconstruction of the depth parameters $\{D_i, D_i'\}$.

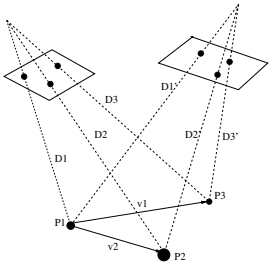

Fig. 11.10 Minimum set of three non-collinear points $\{P_1, P_2, P_3, \dots\}$ in 3D space is used to recover the motion parameters.

The equation (11.11) shows that any translation T changes the lengths $\{D_i\}$ of the associated rays of projection. On the other hand, motion is necessary for the depth reconstruction according to (11.11). We want to recover the motion from observations of a static set of points.

Recursive Algorithm for Simultaneous Motion and Structure Estimation

Since the influence of motion (R,T) and structure D_i is non-linear in

$$D_i' \mathbf{n_i'} = R \cdot D_i \mathbf{n_i} + T, \tag{11.14}$$

therefore, we need to estimate both in parallel. Instead of sampling the rays for the correct solution, we use an algorithm that we originally developed for small positional changes [9, 10], but that proves to be valid for large deviations as well.

The algorithm is based on the idea of alternating refinement of pose and structure information. For small movements in the scene, the assumption is valid that the changes are mostly in the pose parameters (especially rotation R) while the distances to the observed points remain almost the same. Therefore, for each new frame, we start with an initial guess for distances $\{\hat{D}_i^t\}$ that is chosen for each iteration step t as follows:

$$\hat{D}_i^t = \begin{cases} D_{init}, & t = 0 \\ D_i^{t-1}, & t \neq 0 \end{cases} \tag{11.15}$$

These depths are used to calculate guesses for the point positions to

$$\hat{P}'_i = \hat{D}'_i \cdot n'_i \tag{11.16}$$

The initial error for a significant change in pose is depicted in Fig. 11.11 as green(gray) lines and circles. We compute the pose change for these two initial point sets and use it to refine our guess about the depth structure $\{D'_i\}$.

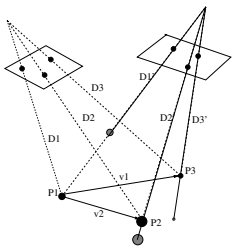

Fig. 11.11 The initial depth assumption for a very large deviation in position.

Computing the absolute orientation is the process of determining $\tilde{R}$ and $\mathbf{T}$ from corresponding pairs $\hat{P}'_i$ and P_i. With three or more non-collinear points, $\tilde{R}$ and $\mathbf{T}$ can be obtained as a solution to the following least-squares problem as described in [15].

$$\min_{R,\mathbf{T}} \sum_{i=1}^{n} \|RP_i + \mathbf{T} - \hat{P}'_i\|^2, \quad \text{subject to} R^T R = I. \tag{11.17}$$

Such a constrained least squares problem [14] can be solved in closed form using quaternions [21, 24], or singular value decomposition (SVD) [20, 2, 21, 24]. We use the SVD method to calculate the rotation matrix in our system as described in [10] in more detail. We see in (11.13) that the rotation alone is sufficient to estimate the changes in the distances to the tracked points. The corresponding translation T can be estimated from the pose change of the corresponding points (P_i, P'_i) assuming that the rotation matrix R is known.

This is an iterative approach, where the result of each iteration is used to estimate new improved guesses of the depth structure D_i.

11.3.3 Open Challenges

The presented VSLAM system was tested in different scales ranging from outdoor navigation down to navigation of endoscopes in medical applications. The system works reliably if the initial depth structure is known. The correct initialization is still an open challenge that we try to approach. The second challenge is a compensation of drifts due to accumulation of errors for the case that the presented system is used

for global localization and the noise values create an offset value deteriorating the localization quality.

11.3.3.1 Estimation of the Initial Depth Relations

The initial depth structure is usually initialized in two ways. In case that the system starts at a known location, like e.g. a landing place or a docking station, a known reference structure can be observed and a reference projection can be calculated from it. As an example, the reference structure can be a rectangle on the floor and the reference view can be a pose with an image plane coplanar to the rectangle with the focal point 1m above the center of the rectangle. Basic projection equations can be used to calculate this "virtual projection". The depth information for a current observation can now be estimated using our iterative approach (Section 11.3.2) in Fig. 11.12.

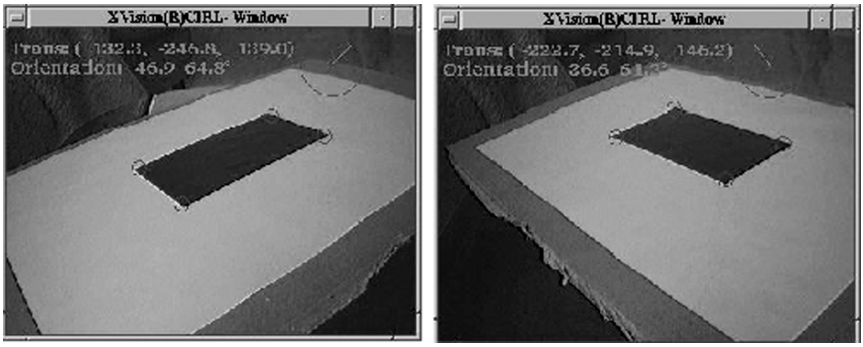

Fig. 11.12 Initialization from a known reference structure.

A second initialization method is based on the essential matrix computation [25]. The motion parameters can be estimated up to an unknown scale in the translation in Fig. 11.13. A relation between the projections p_i, p_i^* in two camera images with known internal parameters can be expressed with the Essential Matrix $\tilde{\mathbf{E}}$ [14] as

$$p_i^* \tilde{\mathbf{E}} p_i = 0 \tag{11.18}$$

The Essential Matrix $\tilde{\mathbf{E}}$ consists of a product of two matrices

$$\tilde{\mathbf{E}} = \tilde{\mathbf{R}} \cdot \mathrm{sk}(\mathbf{T}),$$

$$\text{with} \quad \mathrm{sk}(\mathbf{T}) = \begin{pmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{pmatrix} \tag{11.19}$$

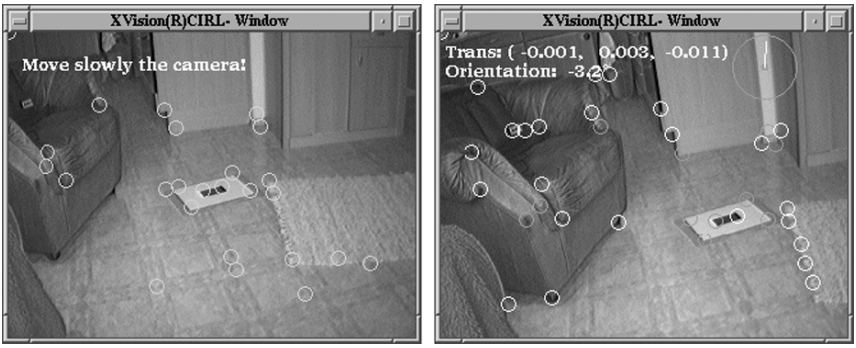

Fig. 11.13 Initialization from an initial motion. After the initial motion a subset of the initial points in the left image can still be tracked. It is used for the essential matrix method.

Note that, given a correspondence, we can form a linear constraint on $\tilde{\mathbf{E}}$. It is only unique up to scale, therefore, we need 8 matches, then we can form a system of the form $\tilde{\mathbf{C}} \cdot e = 0$ where e is the vector of the 9 values in $\tilde{\mathbf{E}}$.

The essential matrix solution gives a valid result only if the corresponding points did not lie on a super-quadric, like e.g. on a plane. This condition occurs unfortunately quite often in case of flying systems observing the ground. For such a configuration a homography matrix method needs to be used. An important decision is to recognize that a given feature set is on a plane without any knowledge about the environment. Our current solution uses the eigenvalues of the essential matrix which should be equal to (1,1,0) in the ideal case. Noise and detection errors cause them to deviate from this ideal case. A planar condition can be identified as a result with two non-zero eigenvalues with a ratio significantly larger than 1. In such a case, the homography solution is chosen.

11.3.3.2 Compensation of Drifts

A typical off-shelf perspective camera has only a limited field of view. Therefore, only a small set of landmarks is usually visible in the sensor cone with an opening angle defined by the focal length of the lens. The shorter the focal length the larger is the field of view of the camera. There are natural limits on the maximum size of the field of view. Fish-eye lenses with a wide field of view have usually significant radial distortions and do not focus in a single point, which deteriorates the quality of the navigation result that relies on the knowledge of the angle of incidence of the light rays.

Our camera model represents the imaged points on a sphere (see Fig. 11.14). They are represented by the normalized direction vectors n_i. This allows us to construct a reference view used for the localization in the local area that spans the entire space. In the initialization, only features contained in the current sensor view

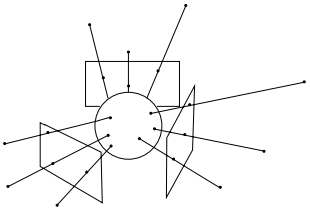

Fig. 11.14 Points in different directions around the origin of the local coordinate frame are getting projected on a sphere and represented as direction vectors n_i.

are used as the initial reference set, but this view gets extended with additional landmarks that are reconstructed using the current motion parameters between images containing a specific landmark. The images used for reconstruction of a specific landmark do not need to contain the initial reference view. The newly estimated landmark position gets transformed back to this initial frame. This allows a reconstruction of landmarks in all directions (360° field of view).

This extension of the field of view permits the usage of one unique set of reference features in a local area independent of the direction of motion. Any noise or error in the localization results in this case just in a noise in the resulting pose estimation. Since the localization is calculated always relative to the same reference structure in the world, we can avoid drifts in the localization that could be caused by continues integration of the relative motions between consecutive frames. The reference frames need to be changed because of the limited range in which a given set can be observed. This hand-off process is an interesting open issue. Interesting solutions can be used from laser based SLAM approaches.

11.4 Results

11.4.1 Convergence of the Pose Estimation

The presented system estimates the pose change between two frames. This can be an incremental change between two consecutive frames or the absolute difference to a reference frame. Depending on the requirements in the system, both modes are of interest. An important question here is the accuracy of the system for varying distances from the original configuration.

The number of iterations required to estimate the motion parameters with an accuracy below 1cm stays below 50 for most large indoor environments tested with this system. The proof of global convergence is mathematically derived in [15]. The number of iterations to find the best transformation explaining the changes between the reference and the current position of the projections varies depending on the initial differences between the reference model and the current pose.

Usually, we don not propagate the changes in the λ_i lengths between the steps. Instead of calculating the change from the reference position, a relative change to the previous step can be calculated which converges in very few ($le10$) iterations.

11.4.2 Reconstruction Results

11.4.2.1 Endoscope Navigation

The system was tested in micro-scale performing pose estimation of the endoscope camera in a phantom of a human skull. The experimental validation of our approach was carried out on the setup depicted in Fig. 11.15.

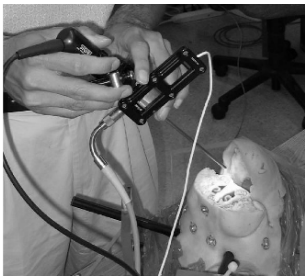

Fig. 11.15 Experimental setup for the validation of the accuracy of the endoscope navigation in a porcine cadaver head.

We tracked the position of the endoscope with the $OptoTrak^{TM}$ system in the background to verify the motion estimation results from our system. The resulting reconstruction errors had a standard deviation of (0.62, 0.3382) for each of the cases. The minimal rotational error expressed as Rodrigues vector was r=(0.0017, 0.0032, 0.0004), (-0.0123, -0.0117, -0.0052) for both cases. The error in the estimate of the translation vector was $\Delta T = (0.05, -0.398, 0.2172)^T, (-0.29, 0.423 - 0.4027)^T [mm]$

11.4.2.2 Outdoor Scene Reconstruction

We used the presented VSLAM system to classify geometric positions of regions in the image to filter candidates for traffic signs. The system was recovering the motion of the camera by tracking of features in the images and performing a 3D reconstruction of the position of the extracted color blobs. Candidates in the right geometric location relative to the road were additionally checked for planarity by adding additional points on the surface in Fig. 11.16.

Fig. 11.16 (Left) Outdoor scene, (right) 3D reconstruction.

11.5 Conclusions

We presented an overview of the navigation approaches tested on our mobile systems. A theoretical background for our approach for explicit recovery of structure and motion from a minimum set of 3 corresponding landmarks in spherical projections was discussed. The presented approach assumes the knowledge about the initial geometrical relation between the depths to the observed points, which may be obtained from a 3D model of the world or from more complex structure-from-motion approaches requiring more points. A good candidate for initialization is, e.g., the eight point algorithm [25] that delivers an initial guess for the depths to the points. This information is refined using the presented 3D-reconstruction. This initial information is updated in the system using a recursive algorithm updating the motion and depth parameters in parallel.

In opposite to other existing approaches, the presented system presents an explicit solution for an arbitrary number of point correspondences in monocular image sequences. We require a minimum of 3 landmarks for the structure and motion recovery, but the system scales easily to more corresponding points, which improve the error compensation capabilities of the system.

Our future work will focus on improvements in the convergence of the system by controlled fixation of parameters depending on the feature configuration and on solving the open challenges mentioned in section 11.3.3.

Acknowledgements The work was partially supported by the DFG Cluster of Excellence in Cognitive Technical Systems (CoTeSys). The author would like to thank Gregory Hager from the Johns Hopkins University, who helped to formulate many of the mathematical foundations for the initial implementation of the presented approaches.

References

1. T. Alter. 3D Pose from 3 Points Using Weak Perspective.. IEEE PAMI, Vol. 16, No. 8, pp. 802-808, 1994
2. K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Trans. Pat. Anal. Machine Intell.*, vol. 9, pp. 698–700, 1987.

3. J. Borenstein and Y. Koren. Real-time Obstacle Avoidance for Fast Mobile Robots in Cluttered Environments. In *IEEE International Conference on Robotics and Automation*, pages 572 – 577, May 1990.
4. D. Burschka, C. Eberst, C. Robl, and G. Färber. Vision-Based Exploration of Indoor Environments. *Workshop on Robust Vision for Vision-Based Control of Motion at the IEEE International Conference on Robotics and Automation*, WS2, May 1998.
5. D. Burschka and G. Hager. Vision-based control of mobile robots. In *Proc. International Conference on Robotics and Automation*, pages 1707–1713, 2001.
6. D. Burschka and G. Hager. Scene Classification from Dense Disparity Maps in Indoor Environments. In *Proc. ICPR*, 2002.
7. D. Burschka and G. Hager. Stereo-Based Obstacle Avoidance in Indoor Environments with Active Sensor Re-Calibration. In *International Conference on Robotics and Automation*, pages 2066–2072, 2002.
8. D. Burschka and G.D. Hager. Vision-Based 3D Scene Analysis for Driver Assistance. *ICRA*, 2005.
9. D. Burschka and Gregory D. Hager. V-GPS – Image-Based Control for 3D Guidance Systems. In *Proc. of IROS*, pages 1789–1795, October 2003.
10. D. Burschka and Gregory D. Hager. V-GPS(SLAM): – Vision-Based Inertial System for Mobile Robots. In *Proc. of ICRA*, pages 409–415, April 2004.
11. D. F. DeMenthon and Larry S. Davis. Model-Based Object Pose in 25 Lines of Code. *International Journal of Computer Vision*, 15:123–141, June 1995.
12. A. J. Davison. Real-Time Simultaneous Localisation and Mapping with a Single Camera. Proc Internation Conference on Computer Vision, Volume 2, pages 1403–1412, 2003.
13. G. Dudek, P. Freedman, and I. Rekleitis. Just-in-time sensing: efficiently combining sonar and laser range data for exploring unknown worlds. *In Proc. of ICRA*, pages 667–672, April 1996.
14. O. Faugeras, *Three-Dimensional Computer Vision*, The MIT Press, 1993.
15. G. Hager, C-P. Lu, and E. Mjolsness. Object pose from video images. *PAMI*, 22(6):610–622, 2000.
16. L.M. Lorigo, R.A. Brooks, and W.E.L. Grimson. Visually-Guided Obstacle Avoidance in Unstructured Environments. *IEEE Conference on Intelligent Robots and Systems*, pages 373–379, September 1997.
17. E. Malis, F. Chaumette, and S. Boudet. 2D 1/2 visual servoing. *IEEE Transactions on Robotics and Automation*, 15(2):238–250, April 1999.
18. R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, 2000.
19. R. Haralick, C. Lee, K. Ottenberg and M. Nölle. Review and Analysis of Solutions of the Three Point Perspective Pose Estimation Problem. International Journal on Computer Vision, 13(3), pages 331–356, 1994.
20. B. K. P. Horn, H. M. Hilden, and S. Negahdaripour, "Closed-form solution of absolute orientation using orthonomal matrices," *J. Opt. Soc. Amer.*, vol. A-5, pp. 1127–1135, 198.
21. B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternion," *J. Opt. Soc. Amer.*, vol. A-4, pp. 629–642, 1987.
22. D. Nister. A Minimal Solution to the Generalised 3-Point Pose Problem. CVPR 2004, 2004.
23. R. C. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. *Autonomous Robot Vehicles*, Springer-Verlag:167–193, 1990.
24. M. W. Walker, L. Shao, and R. A. Volz, "Estimating 3-D location parameters using dual number quaternions," *CVGIP: Image Understanding*, vol. 54, no. 3, pp. 358–367, 1991.
25. E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, 1998.