15

Object Localization using Stereo Vision

Sai Krishna Vuppala
Institute of Automation, University of Bremen
Germany

1. Introduction

Computer vision is the science and technology of machines that see. The theoretical explanation about the optics of the eye and the information about the existence of images formed at the rear of the eye ball are provided by Kepler and Scheiner respectively in the 16th century (Lin, 2002). The field of computer vision is started emerging in the second half of 19th century, since then researchers have been trying to develop the methods and systems aiming at imitating the biological vision process and therefore increasing the machine intelligence for many possible applications in the real world. The theoretical knowledge behind the perception of 3D real world using multiple vision systems has been published in various literature, and (Hartley & Zisserman 2000) (Klette at al., 1998) (Sterger , 2007) are few well known from them. As there are numerous applications of computer vision in service, industrial, surveillance, and surgical etc automation sectors, researchers have been publishing numerous methods and systems that address their specific goals. It is intended with most of these researchers that their developed methods and systems are enough general with respect to the aspects such as functional, robust, time effective and safety issues. Though practical limitations hinder the researchers and developers in achieving these goals, many are getting ahead providing the solutions to the known and predicted problems.

The field of computer vision is supporting the field of robotics with many vision based applications. In service robotics action interpretation and object manipulation are few examples with which the computer vision supports the humans. According to (Taylor & Kleeman, 2006), three things are almost certain about universal service robots of the future: many will have manipulators (and probably legs!), most will have cameras, and almost all will be called upon to grasp, lift, carry, stack or otherwise manipulate objects in our environment. Visual perception and coordination in support of robotic grasping is thus a vital area of research for the progress of universal service robots. Service robotic tasks are usually specified at a supervisory level with reference to general actions and classes of objects. An example of such a task would be: Please get 'the bottle' from 'the fridge'. Here, getting the bottle is the intended action, 'bottle' belongs to the class of objects that are manipulated, and 'fridge' belongs to the class of objects in/on which the objects to be manipulated lie. The content of the manuscript addresses the object manipulation tasks using stereo vision for applications of service robotics. Motivation of the content of the manuscript stems from the needs of service robotic systems FRIEND II and FRIEND III

(Functional Robot with dexterous arm and user-frIENdly interface for Disabled people) that are being developed at IAT (Institute of Automation, University of Bremen, Germany). The systems FRIEND II and FRIEND III are shown in figure 1 a) and b). The objects of interest to be manipulated are shown in figure 2.

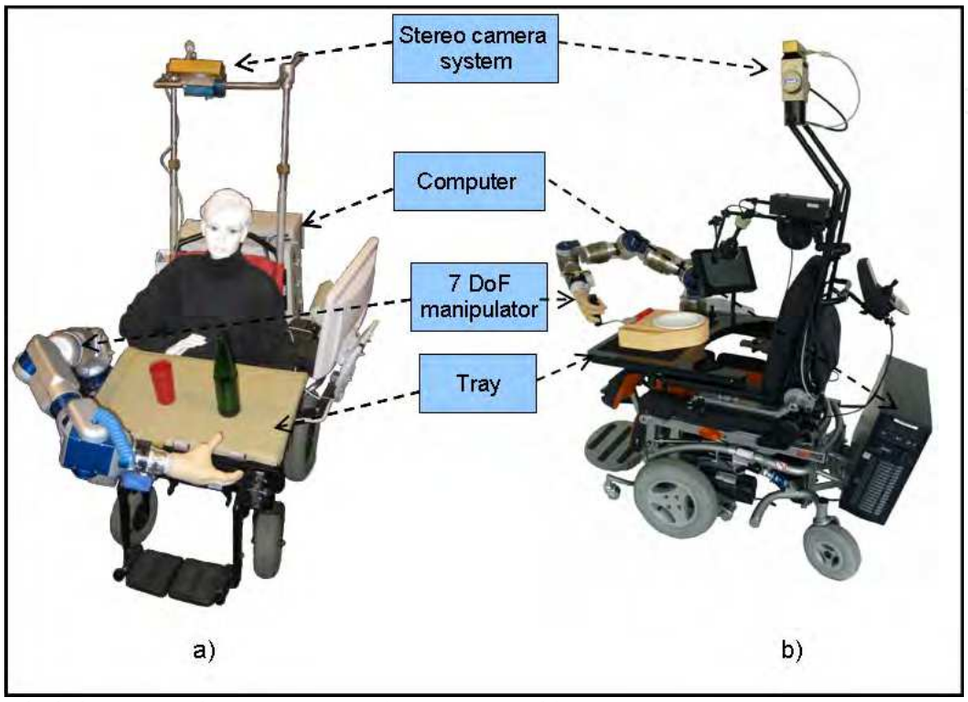


Fig. 1. Rehabilitation robotic systems a) FRIEND II b) FRIEND III

Fig. 2. The objects of interest for the manipulation purpose

In order robot manipulators to perform their actions autonomously, 3D environment information is necessary. In order to reconstruct 3D object information using cameras various approaches have been investigated since few decades. According to (Lange at al., 2006), the 3D object information recovery techniques are broadly classified into passive and active sensing methods. Passive sensing methods require relatively low power compared to the active methods. Passive sensing methods are similar to the biological vision process. The characteristic contrast-related features in images (i.e. cues such as shading, texture) of the

observed scene are used in extracting the 3D information. In active sensing methods compared with passive methods high accuracy measurements can be obtained. Active sensing relies on projecting energy into the environment and interpreting the modified sensory view. CMOS time-of-flight cameras can capture complete 3D image in a single measurement cycle (Lange, 2000), however such a technology is limited with timing resolutions and, the possibility of the 3D information perception depends on the object surface properties.

Considering passive sensing methods, though any number of cameras can be used in 3D vision process, the methods based on stereo vision plays optimal role considering the functional, hardware, and computational issues. Visual servoing system (Corke, 1996) (Hutchinson, 1996) come into this this category. The two major classes of such systems are position- and image based visual servoing systems. In position based visual servoing systems, they are further classified into closed- and open loop systems. Depending on the requirements of the strategy, they can be realized using single, or two cameras. Closed loop systems have high accuracy since the overall system error is minimized using the feedback information, where as in an open-loop system the error depends on the performance of the system in a single step. In contrast to the closed approaches look-then-move is an open loop approach for the autonomous task executions; the vision system identifies the target location in 3D and the robot is driven to the location of interest. In look-then-move approach, the accuracy of the overall robotic system depends on the camera calibration accuracy and the manipulator accuracy in reaching the specified locations in the real world (Garric & Devy 1995). In a system like FRIEND II look-then-move approach is prefered since object manipulation, collision avoidance, and path planning are planned using a standlone stereo camera unit.

Stereo triangulation is the core process in stereo based 3D vision applications, in order to calculate 3D point stereo correspondences information is used. Precisely identified stereo correspondences give precise 3D reconstruction results. Any uncertainty in the result of stereo correspondences can potentially yield a virtually unbounded uncertainty in the result of 3D reconstruction (Lin, 2002). Precise 3D reconstruction additionally requires well calibrated stereo cameras as the calibrated parameters of the stereo view geometry are used in the stereo triangulation process (Hartley & Zissermann 2000). In addition to that 3D reconstruction accuracy further depends on the length of base line (Gilbert at al., 2006), if the base line is shorter the matching process is facilitated, and if the base line is larger the 3D reconstruction accuracy is higher. Approaches for finding the stereo correspondences are broadly classified into two types (Klette at al., 1998); they are intensity and feature-based matching techniques. The state of the art stereo correspondence search algorithms mostly based on various scene and geometry based constraints. These algorithms can not be used for the texture less objects as the reconstructed obejct information is error prone.

The pose (position and orientation) of an object can be estimated using the information from a single or multiple camera views. Assuming that the internal camera parameters are known, the problem of finding the object pose is nothing but finding the orientation and position of object with respect to the camera. The problem of finding the pose of the object using the image and object point correspondences in the literature is described as a perspective n-point problem. The standard perspective n-point problem can be solved using systems of linear equations if correspondences between at least six image and scene points are known. Several researchers provided solutions for this problem considering at least 3

object points. According to Shakunaga in his publication on pose estimation using single camera (Shakunaga, 1991) described that an n-vector body with n >=3 gives at most 8 rotation candidates from object image correspondences. In case if n < 3, the recovery of the rotation of n-vector body is not possible.
The content of the manuscript discusses selection of stereo feature correspondences and determining the stereo correspondences for opted features on texture less objects in sections 2 and 3 respectively; section 4 presents tracking object pose using 2 object points. ; section 5 presents 3D object reconstruction results; Conclusion and References are followed in sections 6 and 7 respectively.

2. Selection of Stereo Feature Correspondences

Scene independent 3D object reconstruction requires information from at least two camera views. Considering stereo vision rather than multiple vision systems for this purpose eases the computational complexity of the system. Stereo vision based 3D reconstruction methods require stereo correspondence information. Traditional problems in finding the stereo correspondence are occlusion, regularity/ repetitiveness. Traditionally these problems are solved using intensity and feature based stereo matching techniques. However, absolute homogeneous surfaces without any sharp features (i.e. edges, corners etc) do not provide proper stereo correspondence information. The domestic objects to be manipulated in the FRIEND environment are some examples of such kind. As the considered object (either each part or the whole body) has uniform color information the intensity based correspondence search methods often provide improper stereo correspondence information. Therefoere, alternatively the edges of the green bottle are observed for the stereo correspondence analysis. Out of all possible edges of green bottle that is shown in figure 3, only the orientation edges of the bottle can be considered for the 3D bottle reconstruction; this is because the stereo correspondence information between other kinds of edges is typically lost.

2.1 Resolutions between 3D and 2D

The 3D reconstruction accuracy depends on, the accuracy of calibration parameters, the sub pixel accuracy of stereo correspondences and the length of the baseline. In addition to these factors, the 3D reconstruction accuracy further depends on the pixel size, the focal length of the camera, and the distance between the camera and the object. In the following the feasible spatial resolution of object surface in the projected scene is analyzed. Figure 4 illustrates the projection of the object surface on to a single image row. The feasible geometrical resolution of the object surface projected on to one row of the image plane is calculated using formula (1). Similarly the feasible resolution of the object projection on to the column of the object plane can be calculated. The width and the height of the pictured scene are also can be approximately calculated multiplying the number of pixels along the rows and columns multiplied with respective feasible resolutions.

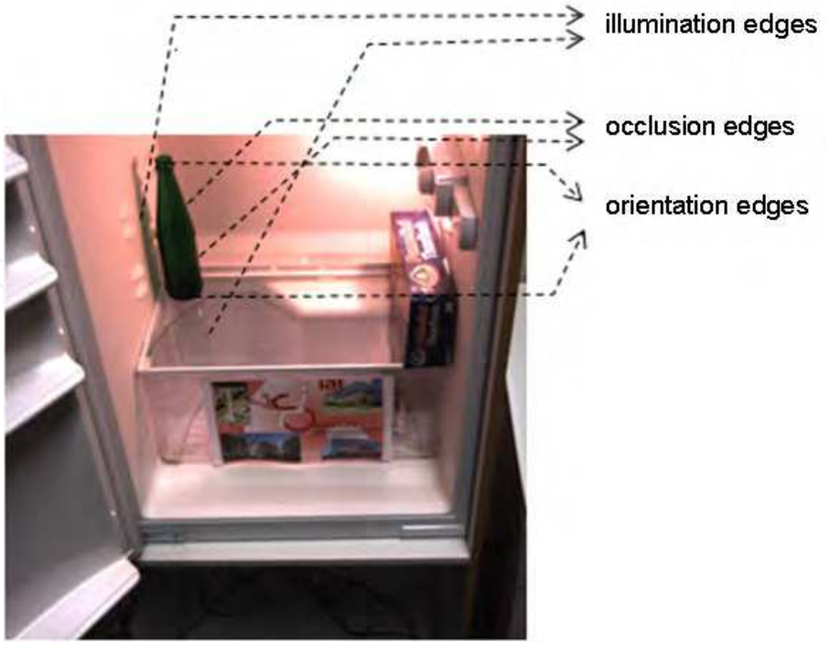


Fig. 3. Various possible edges of green bottle from FRIEND Environment

$$feasible\ resolution = \frac{pd}{f} \tag{1}$$

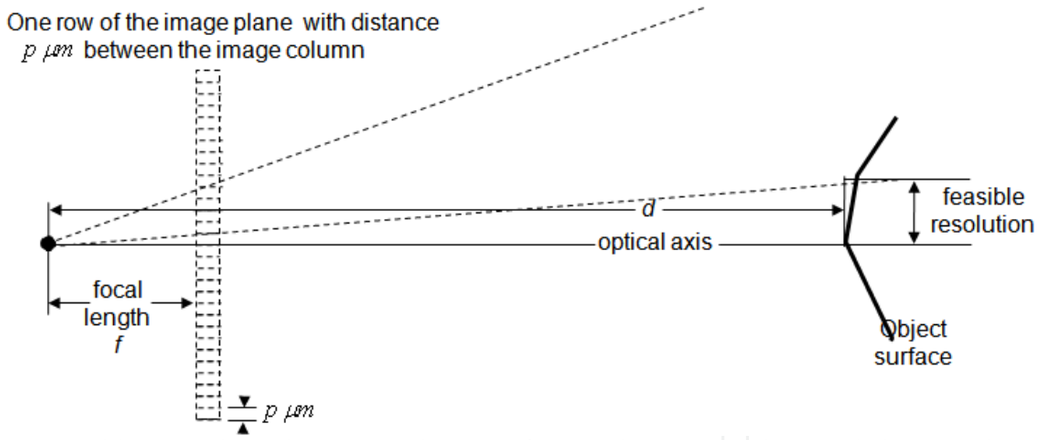


Fig. 4. Feasible spatial resolution on object surface in the projected scene (Klette at al., 1998)

E.g: The size of the SONY CCD Sensor ICX204AK that is used in the bumblebee stereo vision system has the pixel size of $4.54\mu m \times 4.54\mu m$, the focal length of the camera is $4mm$, and the assumed situation of the object is about 1m far from the camera. The feasible resolution of the camera along the row is $1.125mm/\text{column}$ and the feasible resolution along the column is $1.125mm/\text{row}$. Similarly with a higher focal length camera such as $6mm$ the feasible resolution becomes $0.75mm/\text{column}$. Implicitely, with decrease in feasible resolution increases the 3D reconstruction accuracy. From equation (1), the feasible resolution is proportional to the distance between the camera and the object, as the distance

between the object and the camera decreases the feasible resolution for the object decreases and therefore increases the accuracy of 3D reconstruction processes.

2.2 Stereo Feature Correspondences for Homogeneous Object Surfaces

For a pair of matching line segments in stereo images, any point on the first line segment can correspond to every other point on the second line segment, and this ambiguity can only be resolved if the end-points of the two line segments are known exactly. Consider that the line segment AB shown in figure 5 lies on a cylindrical object surface is identified in stereo images. The stereo correspondences for the end points *A* and *B* are considered to be available. As no other external object hides the object line, all the projected object points on line *AB* have corresponding stereo image pixels.

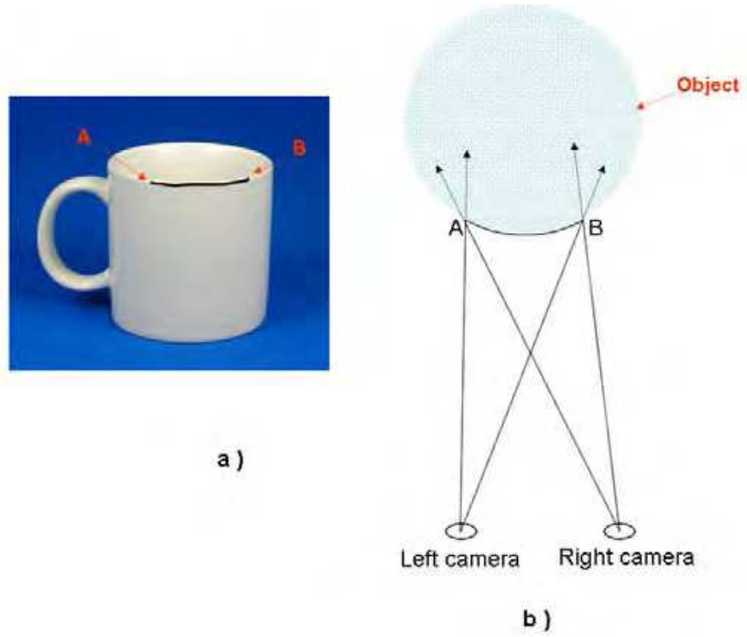


Fig. 5. a) Line segment on the object surface which could be identified in stereo images; b) Significant view of the line segment for the stereo correspondence analysis

It is not always possible to have (or detect) such line segments on the objects whose end points stereo correspondences could be determined. Therefore as an alternative, the points on the object contour (the edges of the object boundary that represent object shape in images) are considered for the stereo correspondence analysis. Surface curvature of an object is inversely proportional to the square of its radius. If the object surface curvature is not equal to zero the visible object contours in stereo images though may appear similar but do not correspond to the same object points. Figures 6a and 6b show the cross sections of object surfaces with different radius of curvatures observed from two different positions, object indexed with 1 has low surface curvature and object indexed with 2 has high radius of curvature. In figures 6a and 6b, the points *A* and *B* are any two points obtained from the contour image of the left camera for object indexed 1, and *A'* and *B'* are the similar points obtained for object indexed 1 from the contour image of the right camera; similarly, points *P* and *Q* are any two points obtained from the contour image of the left camera for object

indexed 2, and P' and Q' are similar points obtained for object indexed 2 from the contour image of the right camera. Though (A,A'), (B,B'), (P,P') and (Q,Q') do not give real correspondences, an approximation of the reconstructed information is not far from the real 3D values. Also, intuitively one can say that the selected points from the high surface curvature give more appropriate stereo correspondences. Therefore if objects do not provide intensity or feature based stereo correspondence information, geometric specific feature points on object surfaces with larger radius of curvatures can be considered in order to have less 3D reconstruction errors. As the considered domestic objects within the FRIEND environment are texture less, approach discussed as above has been used to reconstruct the objects. Figure 7 shows the selection of contour regions in order to have less stereo correspondence errors for the objects that are intended to be manipulated in the FRIEND environment (bottle and meal tray). The contour area marked in the Fig 7a has the high radius of curvature on the surface of the green bottle. In case of the meal tray, as we have considered to reconstruct the meal tray using the red handle (shown in Fig 7b), the marked contour area of the meal tray has the high radius of curvature. Therefore in order to reconstruct these objects feature points are extracted from the marked contour object regions.

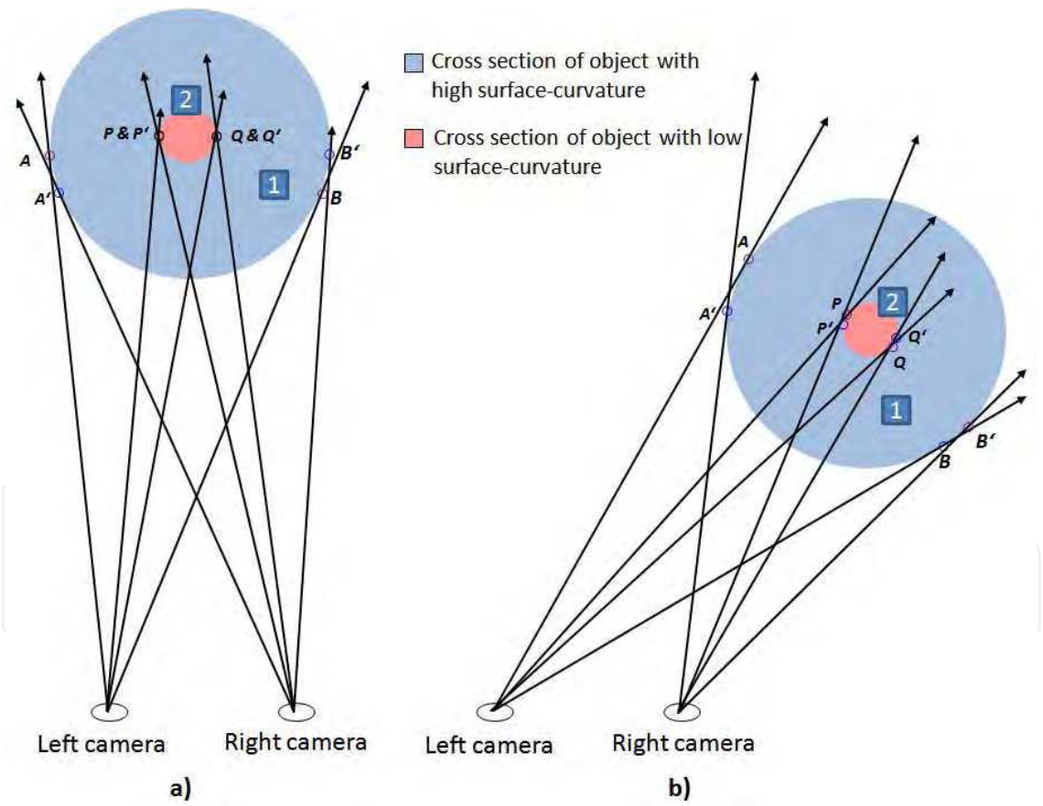


Fig. 6. object surfaces with different radius of curvatures from two different views

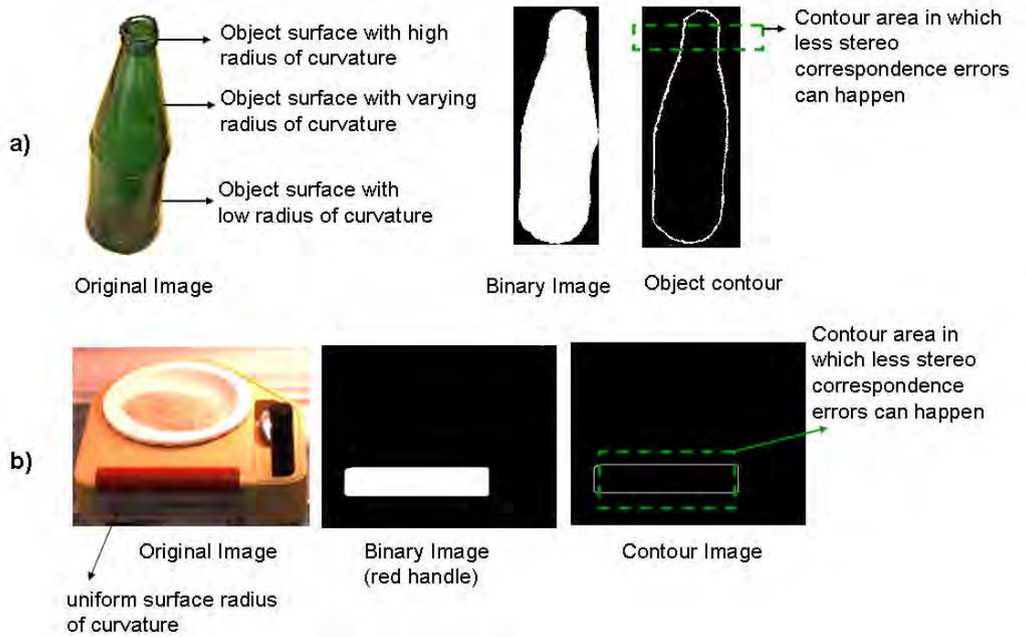


Fig. 7. Images of a bottle and meal tray in different image processing phases

3. Determining the stereo correspondences

As there are foreseen uncertainties exist in the selection of stereo feature correspondences, various constraints have been investigated which give object feature points those could be used for the 3D reconstruction of the object.

3.1 Homography based Stereo Correspondence Calculation

The homography calculated between the stereo images for an intended real world plane can be used for finding the stereo correspondence information. Such a calculated homography between the stereo images must only be used for finding the stereo correspondences for the 3D points that lie on the real world plane considered for the calculation of the homography, otherwise a plane induced parallax phenomena occurs. In order to calculate the homography of the 3D plane at least four stereo correspondences from the stereo image pair are needed (Hartley & Zisserman 2000).

Figure 8 illustrates the plane induced parallax phenomena. The homography induced by the 3D plane π in the stereo images is sufficient for the stereo correspondence calculation of 3D points $U_{\pi 1}$ and $U_{\pi 2}$ those lie on plane π. Whereas, the stereo correspondence for the 3D point U_3 which does not lie on plane π cannot be calculated using H. This is because the mapping $H : u_2 \mapsto u_2'$ is a valid mapping for the 3D point Uπ2 as it lies on plane π, consequently the mapping H always gives u2′ for u2. Therefore the supposed stereo correspondences for the 3D point U_3, i.e. the mapping $H : u_2 \mapsto u_3'$ cannot be done.

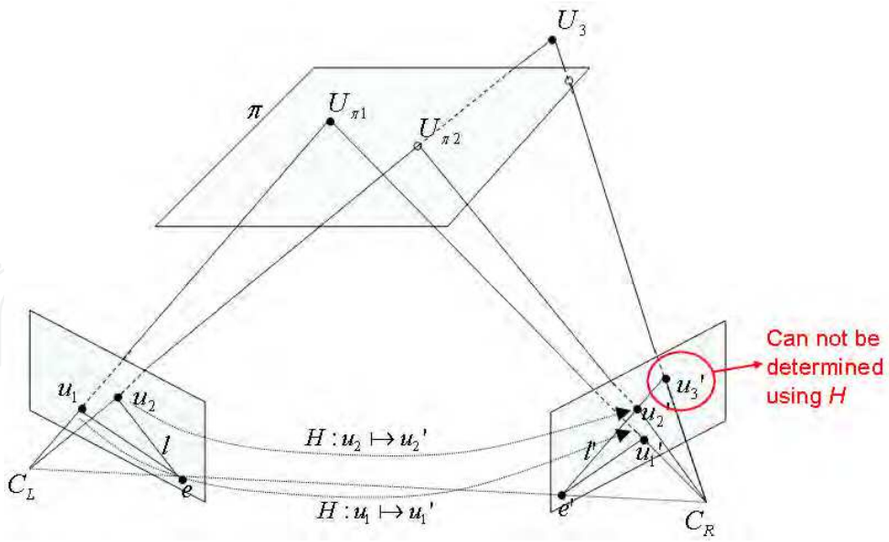


Fig. 8. Sketch: Illustration of plane induced parallax phenomena

The practical results for the plane induced parallax can be clearly seen from figure 9. The homography matrix H between the stereo image pair has been calculated for the chess board π from a real world scene. Fig 9a shows the selected test points on the left image and fig 9b shows the calculated correspondences based on the computed homography. The stereo correspondences for the points u_1', u_2', and u_3' on chess board are computed correctly, the stereo correspondences for the points u_4' and u_5' which are not in the chess board plane are wrongly computed. Therefore it is difficult to use such an approach for the 3D object reconstruction.

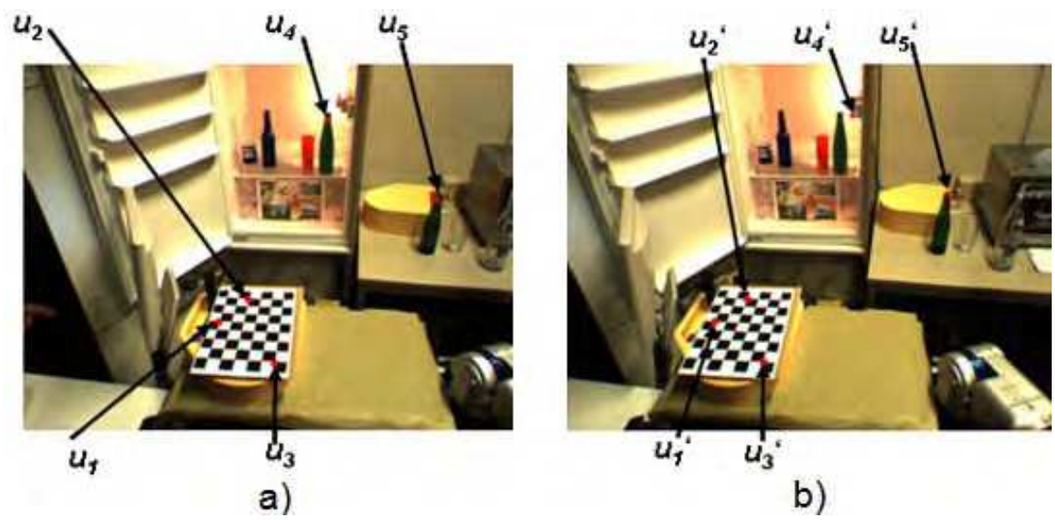


Fig. 9. Calculated stereo correspondences based on homography

3.2 Stereo Correspondences via Disparity Map

In order to find the stereo correspondence information, the algorithm proposed by Birchfield and Tomasi (Birchfield & Tomasi, 1998) has been used. The stereo correspondence

information is obtained using the disparity information obtained from the rectified stereo images. The algorithm improves the previous work in the area of stereo matching. Usually it is difficult to match the pixels of both images when there are big areas without any pattern, this algorithm can handle large untextured regions so the produced disparity maps become reliable. It uses dynamic programming with pruning the bad nodes, which reduces lot of computing time. The image sampling problem is overcome by using a measure of pixel dissimilarity that is insensitive to image sampling. The selection of the parameters of the algorithm is fundamental to obtain a good disparity map. However, the calculated pixel dissimilarity is piecewise constant for textureless objects.

Figures 10, 11, and 12 show the stereo images for 3 different bottle positions and the respective disparity maps. Implementation of the algorithm proposed by the authors of (Birchfield & Tomasi, 1998) is available as open source with OpenCV library. The parameters required for the algorithm are determined offline. In order to test the usability of this algorithm in current context where objects of interest are patternless and are needed to be reconstructed in 3D, a point on the middle of the bottle neck is manually selected on the left image, and the respective right correspondence information is calculated using the disparity value. The stereo correspondences are marked with red circles in the figures; the centres of the circles are the actual positions of the stereo correspondences. In figures 10 and 12, the calculated right correspondence information is approximately in correct position for a human eye, where as in figure 11, the calculated right correspondence information is clearly shifted by several pixels.

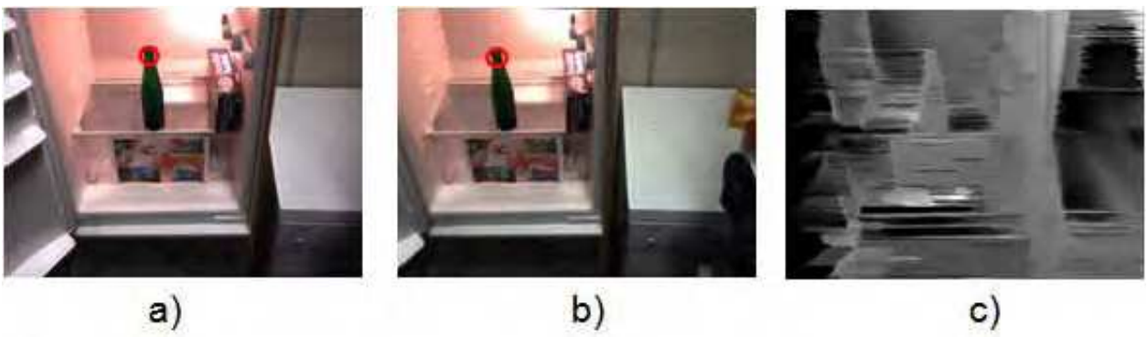


Fig. 10. Disparity map for bottle position 1; stereo images a) Left b) Right; c) Disparity map

Fig. 11. Disparity map for bottle position 2; stereo images a) Left b) Right; c) Disparity map

Fig. 12. Disparity map for bottle position 3; stereo images a) Left b) Right; c) Disparity map

The computation time required for the calculation of the disparity map (1024 * 768 pixels) is approximately 3s on a Pentium computer with 3 GHz processor. As the calculated pixel dissimilarity is piecewise constant for a textureless object, object such as green bottle shown in these figures possess only single, or few gray values, which means the obtained stereo correspondence information is clearly error prone. The quality of disparity map depends also on the selection of the optimal parameters. The calculated stereo correspondence values are presented in section 3.3.2 along with the values obtained using the appoaches discussed in section 3.3.1 and 3.3.2.

3.3 Constrained approaches for 3D point reconstruction

If we know the stereo correspondence information for a 3D object point to be calculated we will have 4 linear independent equations and 3 unknown 3D coordinates of the object point, and therefore we will have an over determined solution to solve the 3D unknown coordinates. In case we do not know anyone of the stereo pair (i.e. either right or left) we will have an under determined solution, i.e 5 unknowns to be solved with 4 equations; or ignoring the equations related with right stereo correspondence result in 3 unknowns to be solved with two equations. Therefore, if we know one of the coordinates from the 3D point (i.e. either X, Y, or Z) the rest coordinates can be calculated either using a single camera, or using a stereo camera system.

3.3.1 Calculation of 3D unknowns using single camera

Let $U=(X, Y, Z, 1)$ is the homogeneous notation for a real world 3D object point to be calculated. Let Z coordinate is known as a priory (i.e. measured or calculated) from U. Let $u=(x, y, 1)$ is the respective image point for U. Let P be the projection matrix (or camera matrix) that relate the image point and real world point for the considered camera. Then, the following cross product holds between the image point and the real world point (Tsai, 1987).

$$u \times PU = 0 \tag{2}$$

we therefore will have two linear independent equations

$$x(p^{3T}U) - (p^{1T}U) = 0 \tag{3}$$

$$y(p^{3T}U) - (p^{2T}U) = 0 \tag{4}$$

where, p_i, i = 1, 2, and 3 is a column vector that represents the ith row of the projection matrix. Using (3) we can have,

$$\Rightarrow X = K_6 + K_7 Y \tag{5}$$

where $K_7 = K_5 / K_4$, $K_6 = K_3 / K_4$, $K_5 = p_{12} - K_{32}$, $K_4 = K_{31} - p_{11}$, and $K_3 = K_2 - K_{33}$, $K_{31} = p_{31}x$, $K_{32} = p_{32x}$, and $K_{33} = xK_1$, $K_2 = p_{13}Z + p_{14}$,. $K_1 = p_{33}Z + p_{34}$. p_{ij} is an element of projection matrix P with an index of i^{th} row and j^{th} column, here i= 1 or 3 and j= 1 till 4. Using (4), we can have

$$\Rightarrow X = K_{15} + K_{14} Y \tag{6}$$

where $K_{15} = K_{13} / K_{11}$, $K_{13} = K_2 - K_{10}$, $K_{12} = p_{22} - K_9$, $K_{11} = K_8 - p_{21}$, $K_{10} = yK_1$, $K_9 = yp_{32}$, and $K_8 = yp_{31}$. Solving (5) and (6) we have the final expression for Y coordinate in (7).

$$Y = (K_6 - K_{15}) / (K_{14} - K_7) \tag{7}$$

As we have Y, the final expression for X can be obtained either from (5) or (6).

3.3.2 Calculation of 3D unknowns using stereo cameras

As the stereo correspondences lie along epipolar lines, the following points are analysed

- For an interested 3D object point in the left image, the values of 3D coordinates X, Y and Z lie along the left projection ray and are unique values for changes in the selection of correspondence point along the right epipolar line
- For an interested 3D object point in the left image, the values of 3D coordinates X, Y and Z lie along the left projection ray are either monotonously increasing, or monotonously decreasing, for one directional changes in the selection of correspondence along the right epipolar line

From these two observations, intuitively one can determine the rest two unknowns if any of the three unknown object point coordinates is known as a priory. In order to consider such an approach a closed loop method is suggested in the following. Gradient descent method or steepest descent method is one of the well known unconstrained optimization algorithms used to find the local minimum of a function $f(x)$. Since the gradient descent method is simple and each of the iteration can be computed fast we have considered this approach for our application. The method starts at an initial state (x_0) in the considered minimization function and moves from a state to its next state (x_i to x_{i+1}) based on the step length (a), the descent direction determined by the gradient (g_i). Selecting a constant step size consumes high computational time in reaching the minimum and also may results in imprecise results, therefore a variable step size rule presented in (Rohn, 1992) is used in our approach. The considered closed loop system is shown in Figure 13. The Z coordinate of the 3D point $U(X,Y,Z)$ is considered to be known and serves as reference input r. The left image point u_L for U is considered to be available. The gradient descent controller computes the right correspondence value u_R along the right epipolar line. The gradient descent method acts on the error value between the reference input and the calculated Z coordinate until the error reaches the tolerance value.

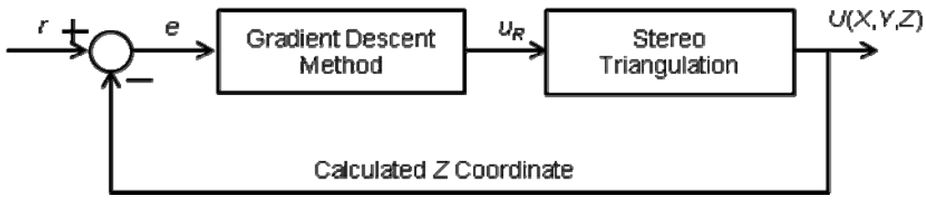


Fig. 13. Closed loop system for precise object localization using gradient descent method

Selected left image point on bottle (in pixels)	Right correpondence and 3D point calculation using method from section 3.2				3D point calculation using method from section 3.3.1				Right correpondence and 3D point calculation using method from section 3.3.2			
	Right pixel	X (m)	Y (m)	Z (m)	Right Correspondence	X (m)	Y (m)	Z (m) known	Right Correspondence	X (m)	Y (m)	Z (m) known
position #1: (412,140)	(280, 140)	0.543	0.543	0.530	Not Applicable	0.525	0.541	0.538	(277.65, 140)	0.521	0.541	0.540
position #2: (495,84)	(389, 84)	0.842	0.471	0.442		0.639	0.468	0.538	(370.424, 84)	0.635	0.468	0.540
Position #3: (239,52)	(120, 52)	0.715	0.727	0.534		0.707	0.726	0.538	(119.232, 52)	0.703	0.727	0.540

Table 1. Comparison of calculated stereo correspondence and 3D information using different approaches

Table 1 presents the values of selected left image points, calculated right correspondences and calculated 3D information for 3 different bottle positions from figures 10, 11 and 12. The calculated 3D information using the constrained methods descussed in section 3.3.1 and 3.3.2 are almost same (i.e. deviation of values is less than 5mm). Where as, the 3D values calucated using the method from section 3.2 for poistion2 are far from the values calcuated using the methods presented in sections 3.3.1 or 3.3.2, position1 has a deviation about 2cm in *X* and less than 5mm deviation in *Y* and *Z* directions, and position 3 values also are deviated less than 1cm. It is observed approach presented in section 3.2 is time consuming and strongly influenced with lighting conditions, texture on object surface, and back ground information etc; the constrained approaches presented in sections 3.3.1 and 3.3.2 have strong limitaion of knowing priory information.

4. Object Pose Estimation using Two Object Points

The algorithm proposed in this chapter supports the "look-then-move" approach in object manipulation tasks. In a "look-then-move" approach, it is required to have 6-Degrees of Freedom (DoF) of object pose in robot base coordinate system assuming the transformation between the robot base and the gripper coordinate system is known to the tolerable precision. The 6-DoF of object pose consists of 3-DoF for the position and the rest 3-DoF for the orientation of the object. The orientation and position of the camera in the object space are traditionally called the camera extrinsic parameters (Yuan, 1989) (Zhang, 2007). The problem of finding camera parameters is described as a perspective n-point problem, since the pose of the object is estimated based on the known set of object points that are identified in the image. Typically the problem is solved using the least squares techniques (Hartley &

Zisseman, 2000) (Lowe, 1987), where a large data set is needed in order to obtain a stable solution. Because of inherent computational complexity of such an approach it is often undesirable in the real-time robotic applications where the actions of the other components of the system depend on the stereo vision output. In order to overcome this problem certain number of researchers has tried to develop methods which require less number of object points for the pose estimation. A summary of the available solutions for pose estimation starting from 3 points is given in (Horaud, 1989). In (Yoon at al., 2003), object 3D pose is tracked using 3 reconstructed distinguishable blobs of an industrial object. The advantage of the algorithm proposed in this chapter is object pose can be tracked using only two object points, therefore it increases the possibility for estimating objects pose which have very less pattern information. Since the algorithm bases on the slope of a line in a plane, vector notation for these two points is not necessary. The constraints of the algorithm are:

- Transformation between the stereo camera system and the considered reference coordinate system is a priory known and kept constant throughout the algorithm execution.
- Objects are not allowed to rotate simultaneously about the multiple axes of the considered reference coordinate system.
- Objects are not allowed to rotate about the virtual line joining the considered two object points, or a parallel line to it in the reference coordinate system.

In a "look-then-move" approach the pose of the object with respect to the robot base has to be known. An example of such a robotic task which belongs to the field of service robotics is to 'fetch the object placed on the platform'. I.e. a compound transformation between the robot base and the ultimate object to be manipulated has to be known. In a next abstraction level the mentioned task can be divided further into two sub tasks, one is to find the pose of the platform with respect to the robot base and the other is to find the pose of the object with respect to the platform. In the first sub task the reference coordinate system belongs to the robot base and in the second sub task the reference coordinate system belongs to the platform. Satisfying the above mentioned constraints, the pose of the platform with respect to the robot base and the pose of the object with respect to the platform can be obtained using the proposed algorithm. In the following, section 4.1 explains the proposed algorithm for tracking the pose of an object, and section 4.2 presents the experimental results.

4.1 Tracking 3D Object Pose using Two Object Points

In this approach two object points are needed to track the object pose, once the object pose is tracked all the other required object points are reconstructed in a reference coordinate system using the priory knowledge of these points in object coordinate system. The algorithm execution is done in the following steps:

- In an initial object situation, the considered two object points are reconstructed in 3D of the reference coordinate system using the calibrated stereo cameras (Hartley & Zisseman, 2000).
- A virtual line joining the considered two 3D points makes an angle with each axis of the reference coordinate system. Using this line, the rotations of object around an axis of the reference coordinate system are tracked up to a span of 180° starting from the initial object situation in the reference coordinate system.

Figure 14 shows the initial situation of the object in 3D. The line joining the 3D points P_1 and P_2 is considered for the algorithm explanation. The orthographic projection of the line on the

xy-plane forms a 2D line $P_1'P_2'$ which makes an angle *a* with *x*-axis. Similarly, the lines $P_1''P_2''$ and $P_1'''P_2'''$ formed in other planes make angles β and γ with *y* and *z* axes respectively.

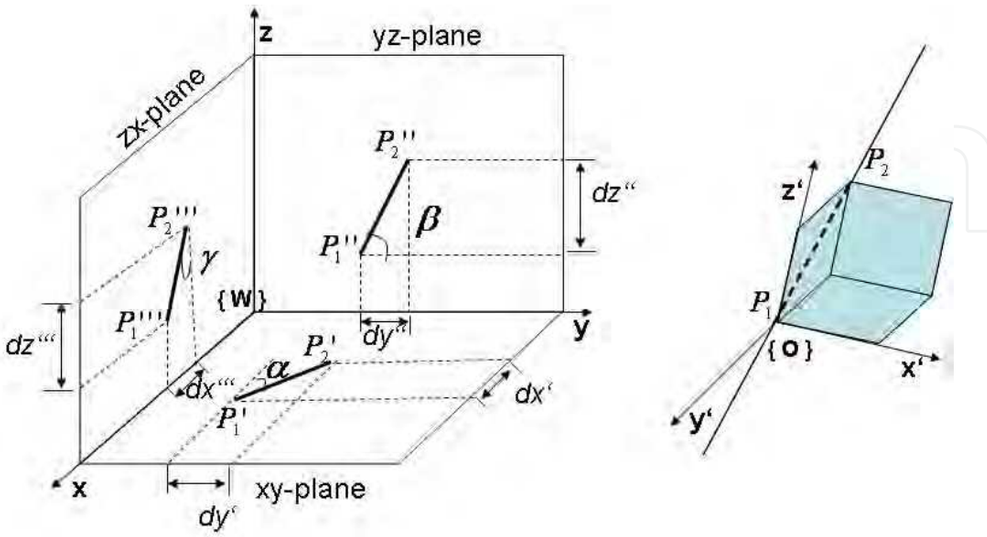


Fig. 14. Orthographic projections of 3D line on each plane of the reference coordinate system that correspond to an initial object situation

$$\alpha = \arctan(dy'/dx') \tag{8}$$

$$\beta = \arctan(dz''/dy'') \tag{9}$$

$$\gamma = \arctan(dx'''/dz''') \tag{10}$$

where dx' and dy' are the lengths of the line $P_1'P_2'$ on the *x* and *y* axes respectively. Similarly dy'' and dz'' are the lengths of the line $P_1''P_2''$ on the *y* and *z* axes, and dx''' and dz''' are the lengths of the line $P_1'''P_2'''$ on the *x* and *z* axes respectively. The ratios dy'/dx', dz''/dy'', and dx'''/dz''' are the slopes of the lines $P_1'P_2'$, $P_1''P_2''$, and $P_1'''P_2'''$ in *xy*, *yz*, and *zx* planes with respect to *x*, *y*, and *z* axes respectively.

4.1.1 Object Rotation around an Axis

When the object is rotated about an axis, the orthographic projections of considered 3D line of the object are changed in all the planes of the reference coordinate system. Figure 15 shows the orthographic projections of the considered object line on the 3 planes of the reference coordinate system when the object is rotated only about the *z*-axis of the reference coordinate system. In order to maintain the transparency, the point of rotation is shown at P_1 in Figure 15, the point of rotation can be any point in 3D. After the rotation, the new rotation angles are a^*, β^*, and γ^*. This information imlicitely tells us that the rotation of object around multiple axes of the reference coordinate system can not be tracked using the slope of the line in reference planes, because, when the object is rotated about multiple axes it has interlinked influence on rotation angles in all the planes.

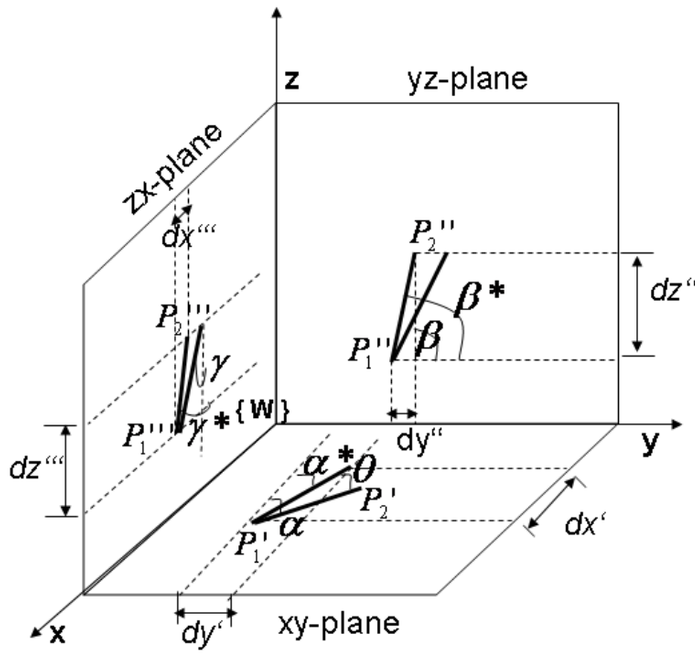


Fig. 15. Rotation of the object around z-axis of the reference coordinate system {*W*}

4.1.2 Calculation of rotation angle

Figure 16 shows the rotations of a line l in xy-plane. The lines l_i and l_i' are the instances of the line l for different slopes, where i = 1, 2, and 3. For each value of i the slope of any line l_i is m_i, and is same for the line l_i'. This information reveals that the slope of the line is same for 180° of its rotation in a plane. Relating this information for the rotations of object in 3D, using two object points object rotation angle around an axis of the reference coordinate system can be tracked uniquely up to a range of 180°. One can customize the range to be only positive or only negative or both positive and negative for object rotations from initial situation.

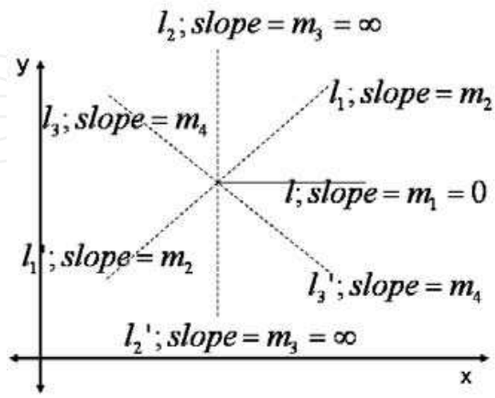


Fig. 16. Slope of the line 'l' for its rotations in xy-plane

In the folllowing, we have considered the object rotation angle (θ) is in a range between -90° and +90° from the object initial situation. In fig 17, the line l is the orthographic projection of

the considered 3D object line in the *xy*-plane, it makes an angle α with the *x*-axis in its initial object situation. Line *l** is the orthographic projection of the considered 3D object line when the object is rotated around the z-axis of the reference coordinate system for an angle θ. The line *l** makes an angle *a-Current* with the *x*-axis. The angle θ can be calculated using (11) and (12).

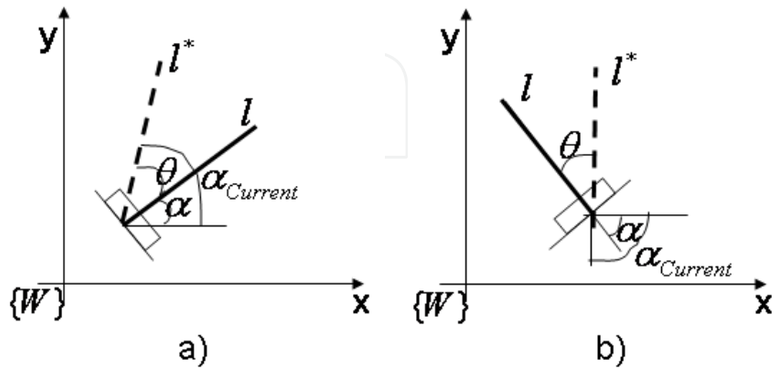


Fig. 17. Orthographic projection of 3D object line in the *xy*-plane; line *l* makes a) positive inclination (*a* >=0) and b) negative inclination (*a* < 0) in the *xy*-plane with *x*-axis

$$if\ \alpha \geq 0,\ \theta = \begin{cases} \alpha_{Current} - \alpha; if\ \alpha_{Current} > \alpha_{NegativeLimit} \\ \alpha_{Current} - \alpha + 180°; if\ \alpha_{Current} <= \alpha_{NegativeLimit} \end{cases} \tag{11}$$

$$if\ \alpha < 0,\ \theta = \begin{cases} \alpha_{Current} - \alpha - 180°; if\ \alpha_{Current} > \alpha_{PositiveLimit} \\ \alpha_{Current} - \alpha; if\ \alpha_{Current} <= \alpha_{PositiveLimit} \end{cases} \tag{12}$$

where, $\alpha_{NegativeLimit} = \alpha - 90°$ and, $\alpha_{PositiveLimit} = \alpha + 90°$

4.1.3 Position of object point in reference coordinate system

Figure 18 shows rotation of an object around *z*-axis of the reference coordinate system. The position of an object point P_3 is initially known in object coordinate system and is intended to be reconstructed in a reference coordinate system. According to the previous discussion, the line *l* joining the two object points P_1 and P_2 tracks the object rotation angle around *z*-axis up to a range of 180° from object's initial situation. When the object is rotated about *z*-axis of the reference coordinate system, the position of a point P_3 is estimated in the reference coordinate system using (13) and (14).

$$^{W}p_{P_3} = {}^{W}_{O'}T\ {}^{O'}p_{P_3} \tag{13}$$

$${}^{W}_{O'}T = \begin{bmatrix} rot(z,\theta) & {}^{W}p_{O'} \\ 0 & 1 \end{bmatrix} \tag{14}$$

where, $\theta = \angle(l, l^*)$ is obtained from stereo vision using (11) and (12). {*O*} and {*O'*} denote the object origins in initial situation and when the object is moved respectively. The coordinate system {*W*} denotes the reference coordinate system. $^{O'}p_{P_i}$ is same as $^{O}p_{P_i}$, and denotes

the priory known position of the object point P_i , where i = 1,2, and 3, in object coordinate system. ${}^{W}p_{P_3}$ denotes the position of the point P_3 in the reference coordinate system. ${}^{W}p_{O'}$ is the position of the new object origin in the reference coordinate system, and in the considered case it is same as ${}^{W}p_{P_1}$. ${}^{W}p_{P_1}$ denotes the position of point P_1 in the reference coordinate system.

4.2 Experimental Results

In order to test the algorithm accuracy two small blobs as features are used on the meal tray. The proposed algorithm is tested for the rotations of the meal tray around z-axis of the reference coordinate system. The origin of the reference coordinate system is located about 90cm far from the middle point of the stereo vision system. The presented object rotations are ranged "from -80° till 80° in steps of 10°" around z-axis in the reference coordinate system from an initial object situation. Since object rotations are around the z-axis the rotation angle is tracked in xy-plane of the reference coordinate system. Figure 19 shows, calculated object rotation angle using the proposed algorithm versus the manually measured rotation angle, when the object is rotated about z-axis of the reference coordinate system. The Pearson correlation coefficient is a measure of extent to which two random variables are linearly related. The Pearson correlation coefficient for the calculated rotation angle and the manually measured rotation angle is 0.999, which means that they are strongly and positively correlated.

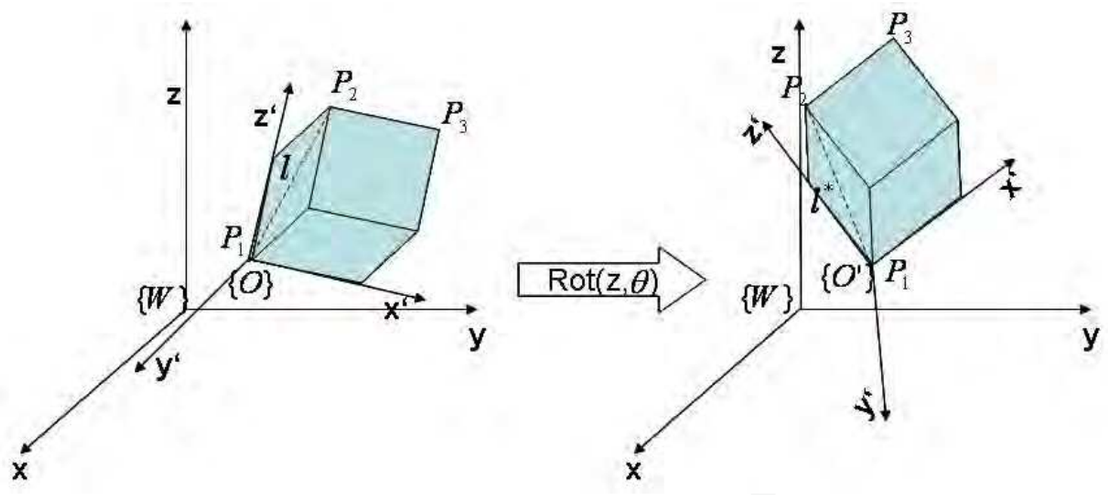


Fig. 18. Rotation of an object around z-axis of the reference coordinate system

Figure 20 shows the results for the reconstruction of an object point P_3 in the reference coordinate system, the position of the considered point in object coordinate system is priory known. Horizontal axis shows reconstructed 3D point P_3 using direct stereo correspondences. Vertical axis shows the calculated 3D point using the proposed algorithm with equations (13) and (14). The Pearson correlation coefficients for the x and y values of the calculated 3D point using direct stereo correspondences and proposed algorithm are 0.9992 and 0.9997 respectively, which clearly shows that they are strongly and positively

correlated. Considering the Pearson correlation coefficient for the z value is not meaningful since it is always pointed to a constant value in the reference coordinate system for the considered case. Therefore the z-value is measured here with the standard deviation of the errors between the two methods, and is 0.005m.

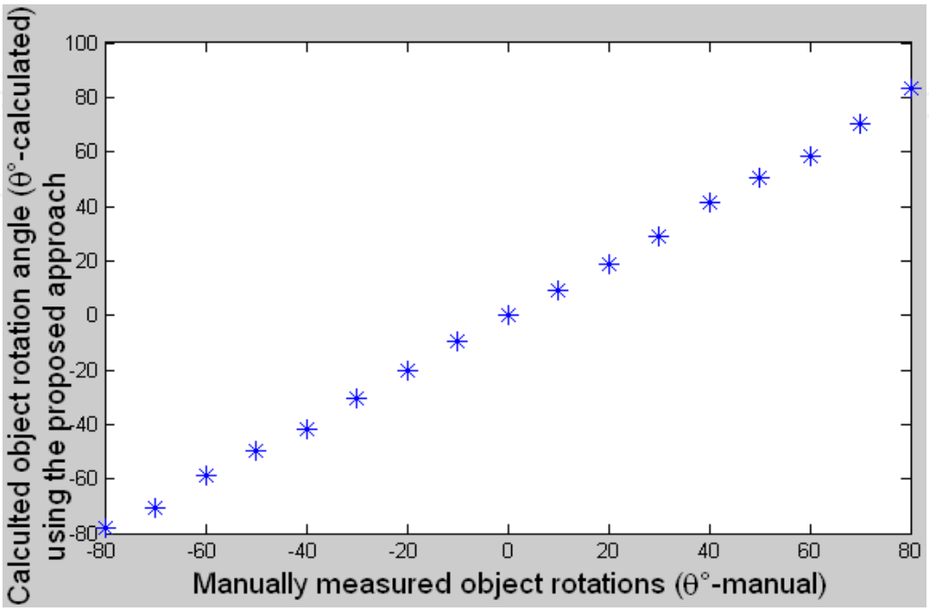


Fig. 19. Calculated rotation angle (θ°-calculated.) using proposed approach versus manually measured rotation angle (θ°-manual)

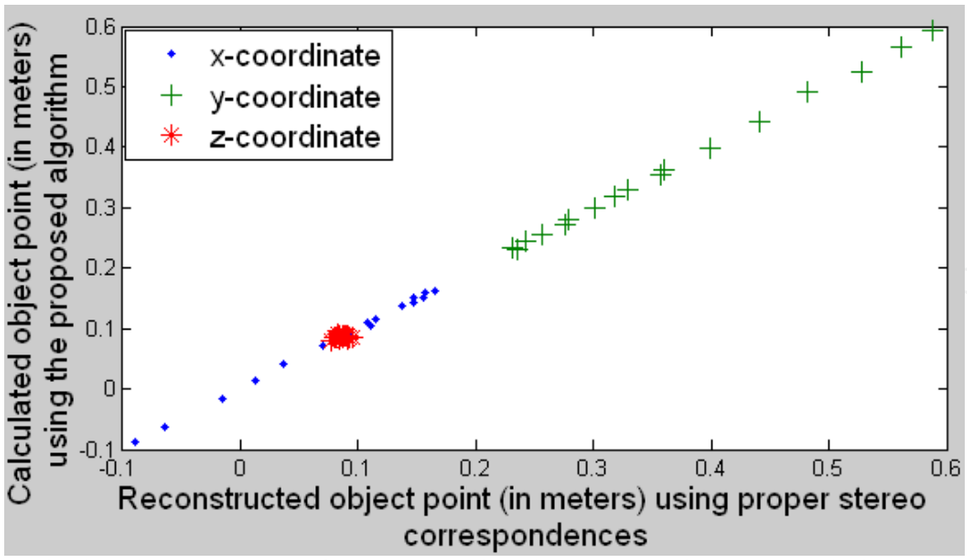


Fig. 20. Calculated 3D position of a point on the object using the proposed algorithm versus reconstructed position of the same point using direct stereo correspondences

5. 3D reconstruction results

In system FRIEND II stereo vision provides 3D object pose information in various scenarios and two of them are "serving drink" and "serving meal". In serving drink scenario, bottle which is available in a refrigerator is fetched and the drink is poured into a glass that is placed on a smart tray, and finally the drink is served it to the user. In the serving meal scenario meal tray is taken from a refrigerator and is taken into a microwave, after getting meal warmed it is served to the user. In order to have synchronised communication among the various sensors of the robotic system the locations of the objects within the service robotic environment are described in a common coordinate system called world coordinate system. In both the mentioned scenarios objects of interest are needed to be precisely localized. In order to determine the transformations between stereo camera and big objects such as refrigerator and microwave special patterns (e.g. chessboard or AR-Marker) are used. Considering the coordinate systems of big objects as reference coordinate systems, meal tray and bottle poses in the FRIEND II environment are calculated. The 3D models of all objects in the FRIEND II environment are predefined and are used in path planning and collision avoidance tasks. In order to reconstruct the 3D object models (bottle and meal tray), feature points are selected from the surface regions that are suggested in section 2.2.

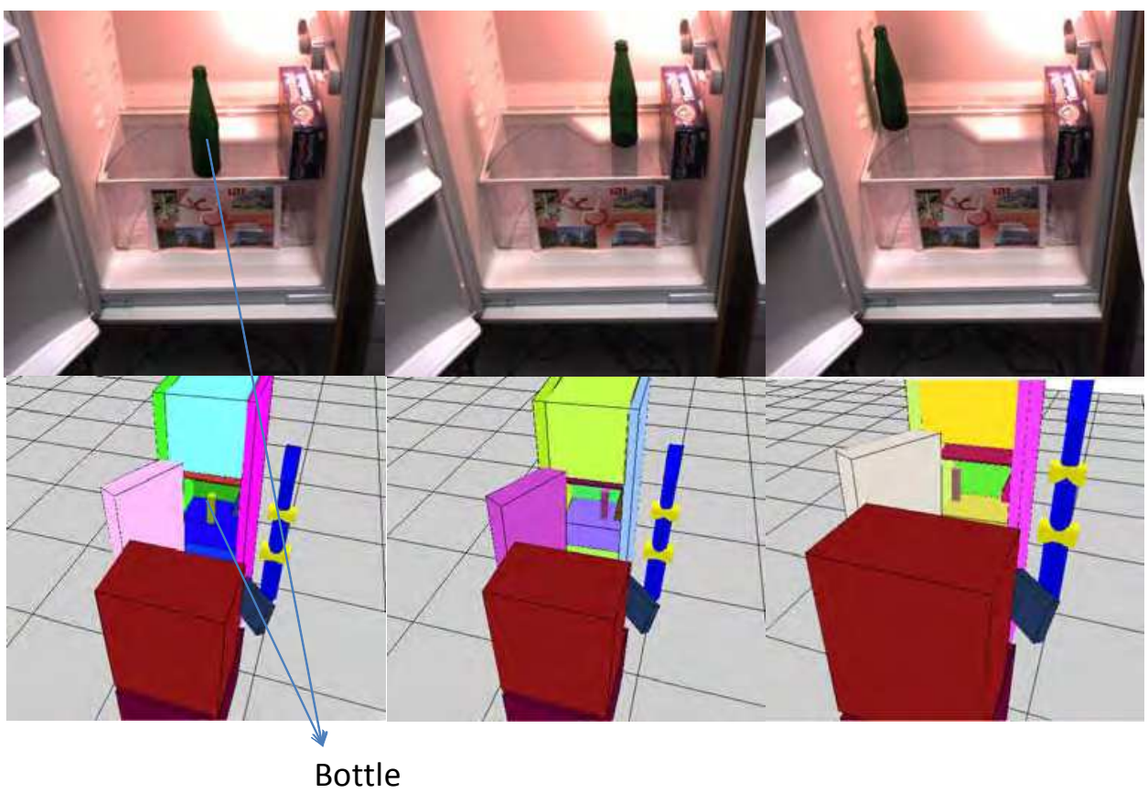


Fig. 21. Top row shows stereo left images of bottles inside a refrigerator; bottom row shows reconstructed bottle (cylindrical object) inside a refrigerator.

In the considered serving drink scenario it is expected that bottle always placed straight on a planar surface (i.e. either on a platform or on a shelf of the refrigerator). Therefore, in order

to reconstruct the 3D model of the bottle only the position of bottle on the considered platform is required. Therefore in order to reconstruct the bottle position, the midpoint from the neck of the bottle is used as the feature point. The stereo correspondences are calculated using the method discussed in section 3.2. Three reconstructed positions of the bottles inside a refrigerator are shown in figure 21. The 3D model of the refrigerator is reconstructed based on the pattern information attached to the refrigerator; the 3D model of the robot arm is reconstructed using the position information provided by the encoder present in each joint. Small cylindrical object shown in the refrigerator model is the reconstructed 3D model of bottle. Numerous times the scenario has been successfully executed and is presented in international fairs ICORR 2007 at Noordwijk Holland, and in CeBIT 2008 at Hannover Germany. In case of no detection of the bottle or if the bottle presents outside the work area of the scenario execution regions, user is warned or informed accordingly. Due to the surface uniformity bottles used in FRIEND II could be grasped in any direction perpendicular to the principal axis.

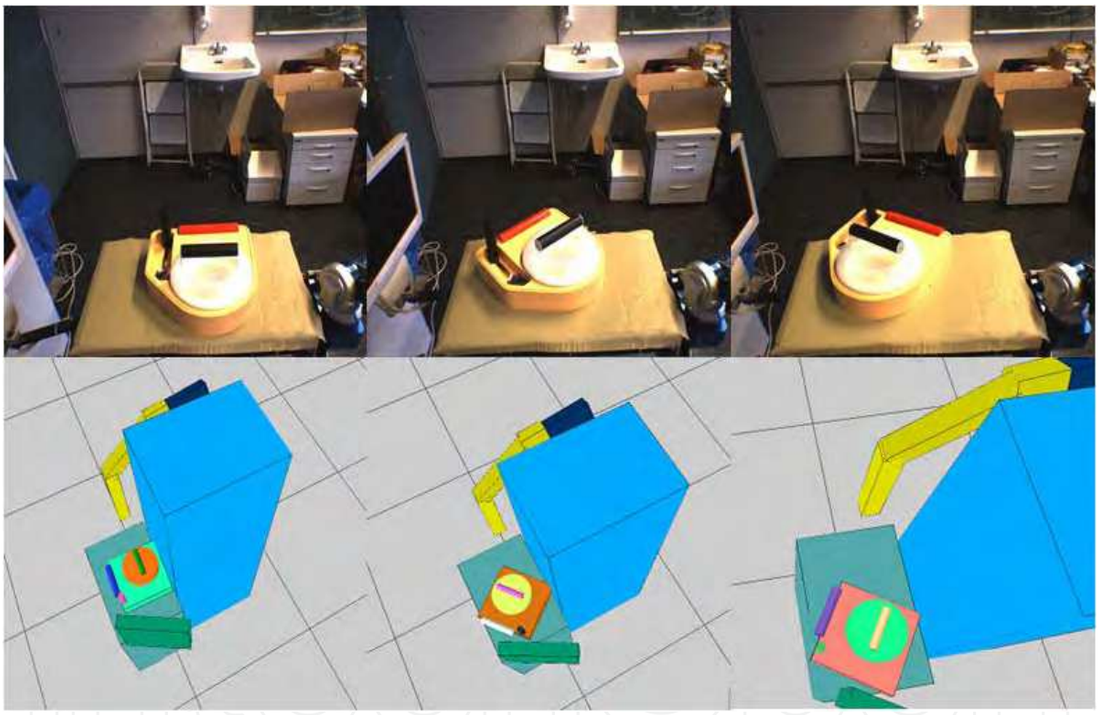


Fig. 22. Reconstructed meal tray poses from meal serving platform

Whereas meal trays used in FRIEND II environment could be grasped in only direction, i.e. a perpendicular direction to the principal axis of the meal tray handle. All components (spoon, meal cover, meal base, cover handle, meal plate) poses have to be precisely estimated in order robot manipulator serve meal to the user securely. As it can be observed from the figure 22, meal tray has no pattern information and therefore finding the stereo correspondence information and consequently 3D reconstruction becomes difficult. During the meal serving scenario, meal tray is available on a planar surface, i.e. either on the shelf of the refrigerator or inside a microwave or on the meal serving platform. Therefore the pose of meal tray on a planar surface only has 4 DoF (i.e. 3 DoF for the position of meal tray on the

plane and 1 DoF for the rotation). Considering robot arm reaching positions, meal tray rotations are further restricted up to 180°. The rotation angle of meal tray is determined using two defined feature points of the red handle. The stereo feature points are extracted from the surface region of the red handle described in section 2.2.

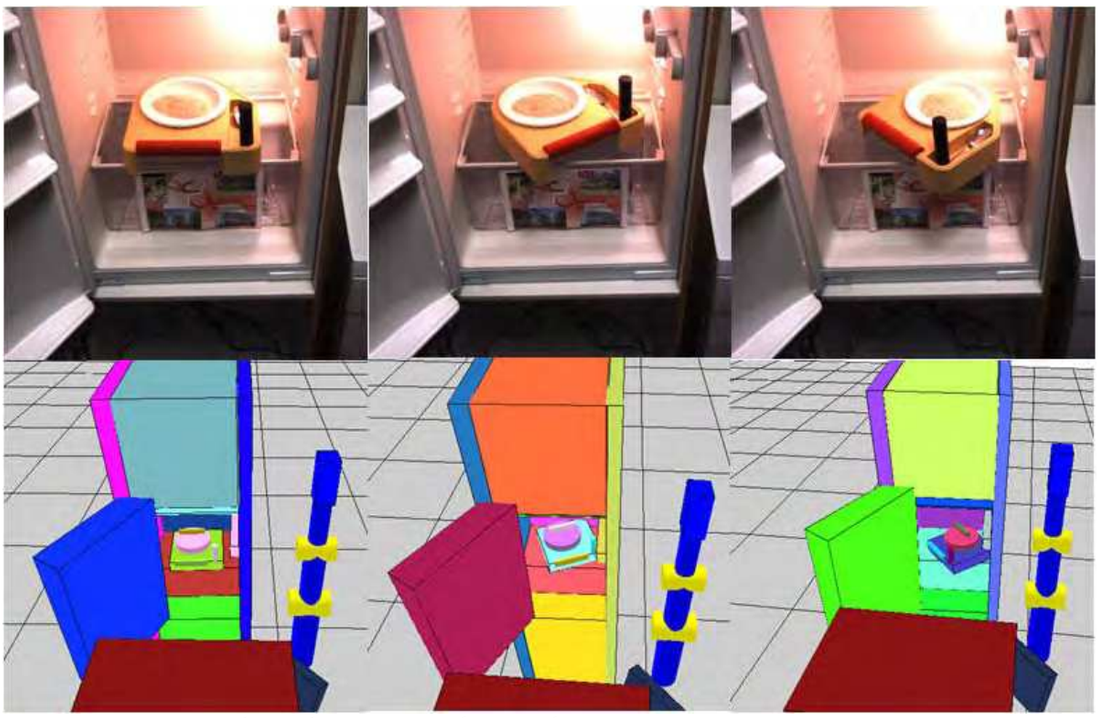


Fig. 23. Reconstructed meal tray poses from refrigerator

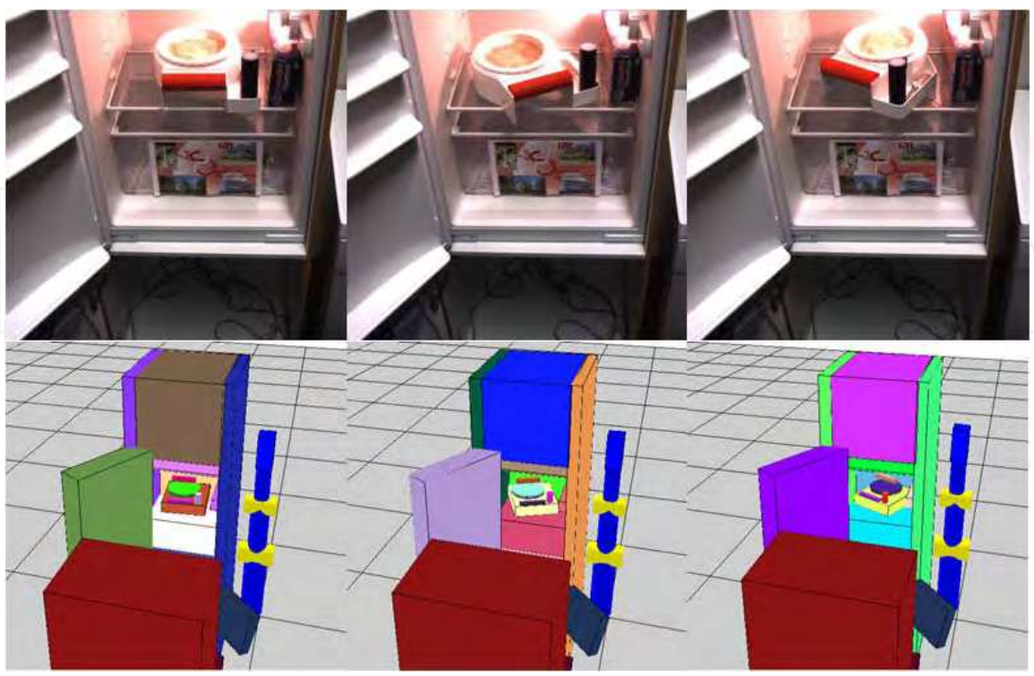


Fig. 24. Reconstructed meal tray poses from refrigerator

The binary image of the red handle represents a solid rectangle; the two end points are the mid points of short edges of minimum fitting rectangle. The rotation angle of the meal tray is determined according to the tracking method described in section 4.
Figures 22, 23 and 24 show the reconstructed meal tray images; top rows show the left stereo images and bottom images show reconstructed meal tray images. The meal tray shown in figure 24 is different from the one shown in figures 22 and 23. The meal serving scenario has been successfully executed numerous times at IAT and is presented in FRIEND III project meetings.

6. Conclusion

In autonomous object manipulation tasks, image processing and computer vision systems play vital role as the objects of interest shall be identified and localized autonomously. As it is always not possible to have calibration targets on the objects of interest that are needed to be localized and manipulated, special object recognition and localization methods which can be applied to such objects are required. The examples of such objects are domestic objects of our daily lives. The FRIEND system that is being developed at IAT supports the physically challenged people in their daily living scenarios such as serving a drink and serving a meal. The domestic objects used in FRIEND system belong to the category of texture less objects. In order to reconstruct the 3D models of these objects stereo vision based 3D object model reconstruction methods are investigated in this thesis work.
In order to reconstruct 3D object models selection of stereo feature points is required; as false stereo correspondences can result in large 3D reconstruction results, selection of stereo feature points for the texture less objects becomes a critical task. In case traditional stereo matching (i.e. intensity and feature based) techniques fail in finding stereo correspondences, geometry specific stereo feature points from high surface curvature regions is selected, stereo correspondences or unknown 3D coordinates information can be retrieved using constrained approaches which require priory object (or feature point in 3D) knowledge. Though strong limitations exist in the proposed tracking method, in contrast to the traditional methods object pose can be tracked using two object points; several applications could be benefited with such an approach. Serving meal and serving drink scenarios of FRIEND system have been successfully realised and currently being presented in international fairs using discussed stereo vision based object localization methods.

7. References

Birchfield.S & Tomasi.C. (1998). *Depth discontinuities by pixel-to-pixel stereo*, In: proceedings of 6th IEEE Int. Conf. Computer Vision, Mumbai, India, pp 1073-1080.

Corke. P. I. (1996). *Visual Control of Robots*. Research Studies Press Ltd, ISBN 0863802709, U.K

Garric. V & Devy. M, (1995). *Evaluation of the Calibration and Localization Methods for Visually Guided Grasping*, In: International Conference on Intelligent Robots and Systems, pp. 387-393. IEEE Press, Pittsburgh

Gilbert. S, Laganiere. R, Roth. G. (2006). Robust Object Pose Estimation from Feature Based Stereo, *IEEE Transactions on Measurement*, Vol 55, Number 4, NRC 48741

Hartley. R & Zisserman. A. (2000). Multiple View Geometry in Computer Vision, Cambridge Univ Press, ISBN 0521623049 ,UK

Horaud. R, Conio. B, Leboulleux. O & Lacolle. B. (1989). *An Analytic Solution for the Perspective 4-Point Problem*, In: Computer Vision and Pattern Recognition, pp 500—507. IEEE Press, San Diego

Hutchinson. S, Hager. G & Corke. P. (1996). A Tutorial on Visual Servo Control. *IEEE Transactions on Robotics and Automation*. Vol. 12, 651—670.

Klette.R , Schlüns.K & Koschan. A. (1998). *Computer Vision: Three-Dimensional Data from Images*. Springer, ISBN 9813083719, Singapore.

Lange. R. (2000). *3D Time-of-Flight Distance Measurement with Custom Solid-State Image Sensors in CMOS/CCD-Technology*, Ph.D. Thesis, ETH-Zürich.

Lange.R, Seitz. P, Biber. A, Schwarte. R. (1999). *Time-of-flight range imaging with a custom solid-state image sensor*, In: Proceedings of the SPIE, Vol. 3823, pp. 180-191, Munich.

Lin, M. (2002). *Surfaces with Occlusions from Layered Stereo, PhD thesi*s, Stanford University, California, United States.

Lowe. D. G. (1987) . Three-dimensional Object Recognition from Single Two-dimensional Images, *Artificial Intelligence*. Vol. 31, Number 3, pp: 355—395, ISSN:0004-3702

Rohn. J. (1992). Step size rule for unconstrained optimization, Vol 49, Number 4, Springer Wien, ISBN 0010-485X (Print) 1436-5057 (Online),

Shakunaga.T. (1991). `Pose estimation of jointed structures, In: Proceedings of Computer Vision and Pattern Recongnition, pp.566-572, Maui, USA.

Sterger. C, Ulrich. M & Wiedermann. C. (2007). *Machine Vision Algorithms and Applications*, WILEY-WCH, ISBN: 9783527407347, Germany.

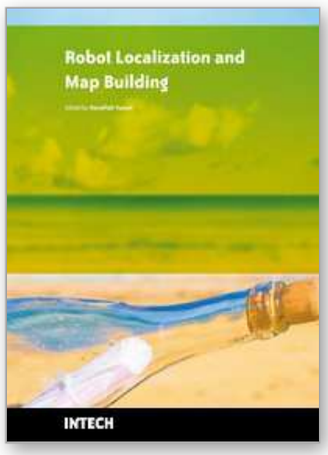
Taylor. G. R & Kleeman. L. (2006) . *Visual Perception and Robotic Manipulation*, Springer-Verlag GmbH, ISBN: 9783540334545.

Tsai. R.Y. (1987): A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses, *IEEE Int. Journal Robotics and Automation*, Vol. 3, Number 4, pp. 323-344.

Yuan. J.S.C. (1989). A General Photogrammetric Method for Determining Object Position and Orientation, *IEEE Transactions on Robotics and Automation*, Vol 5, Number 2, pp: 129—142

Zhang, Z. (2000). A flexible New Technique for Camera Calibration, IEEE Transactions Pattern. Analysis and Machine Intelligence, Vol . 22, Number 11, pp: 1330-1334.

Yoon. Y, DeSouza. G. N, Kak A. C. (2003). *Real-time Tracking and Pose Estimation for Industrial Objects using Geometric Features*. In: Proceedings of the Int. Conference in Robotics and Automation, pp. 3473-3478. IEEE Press, Taiwan



Robot Localization and Map Building

Edited by Hanafiah Yussof

ISBN 978-953-7619-83-1
Hard cover, 578 pages
Publisher InTech
Published online 01, March, 2010
Published in print edition March, 2010

Localization and mapping are the essence of successful navigation in mobile platform technology. Localization is a fundamental task in order to achieve high levels of autonomy in robot navigation and robustness in vehicle positioning. Robot localization and mapping is commonly related to cartography, combining science, technique and computation to build a trajectory map that reality can be modelled in ways that communicate spatial information effectively. This book describes comprehensive introduction, theories and applications related to localization, positioning and map building in mobile robot and autonomous vehicle platforms. It is organized in twenty seven chapters. Each chapter is rich with different degrees of details and approaches, supported by unique and actual resources that make it possible for readers to explore and learn the up to date knowledge in robot navigation technology. Understanding the theory and principles described in this book requires a multidisciplinary background of robotics, nonlinear system, sensor network, network engineering, computer science, physics, etc.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen