Book Detection and Grasping in Library Scenario

Stefan Heyer, Bashar Enjarini, Christos Fragkopoulos and Axel Graeser
Institute of Automation (IAT), University of Bremen, Otto-Hahn-Allee 1, 28359 Bremen,
{sheyer, enjarini, cfragkopoulos, ag}@iat.uni-bremen.de

Abstract

Total or shared autonomous behavior is still an open challenge for service robots. An example of such system is FRIEND (**F**unctional **r**obot arm with user-fr**ien**dly interface for **D**isabled people) (**Figure 1**). In the project ReIntegraRob care robot FRIEND will carry out all necessary manipulations for a completely paralyzed person who with support of FRIEND will be able to work in library. The task of the robot is to autonomously detect books on a shelf, grasp them and place them on a book holder and later place them onto the shelf again. The environment is cluttered since it includes book cart, books and book holder. The books have unknown size and color and no markers which could be helpful in detecting them. The 7 Degrees of Freedom robot arm receives environment information from overhead stereo and gripper mounted hand camera. The paper describes new approaches for detecting and grasping the book reliably. The proposed approach combines two algorithms for book detection and grasping and uses stereo vision together with hand camera to achieve a high rate of success.

Keywords: Visual servoing, book detection, library scenario, planar segmentation, motion planning, grasping

1 Introduction

With the support of FRIEND a paralyzed person (quadriplegia) will turn back to professional life performing the tasks of a librarian without being dependent on personal assistance. The assistant robot FRIEND is going to be used for that purpose [10]. As shown in **Figure 1** the task scenario consists of the FRIEND robot, the book cart where the books are placed on the upper shelf of the cart and the book holder. The task of the robot is to grasp the right most book from upper shelf and place it on the book holder. The pages of the book will be flipped using a device, which is under development. The user reads and inserts the book information (i.e. title, year and publisher etc.) into the library database through a speech recognition program. The robot then grasps the book from the book holder and returns it back to the lower shelf of the book cart. One challenge of this project is to detect, grasp and place books on a book holder or a shelf with a high rate of success (i.e. 99.9%) in order to reduce the interaction between the user and the robot. Size, color and position of the books are unknown and they are not marked. This work focuses on detecting and grasping the right most book on the upper shelf. The reliable detection and position determination of the book on the shelf proved to be astonishingly challenging although book detection and differentiation between several books even with same unstructured cover is an easy task for a human. Both the book cart and the book holder are equipped with markers to allow identification and determination of the 3D position of the book cart and book holder. This supports path planning and collision avoidance. A specific arrangement of the books on the shelf (alternating portrait and landscape placement and book front to the robot) is requested to ease grasping and positioning of the book at book holder and avoiding construction of a specific gripper (see **Figure 1** and **Figure 2**). An industrial parallel gripper is used in this project. A horizontal black stripe is fixed on the rearward vertical board of the book cart to support visual servoing and for validation purpose (see Section 2.3 and Section 3).

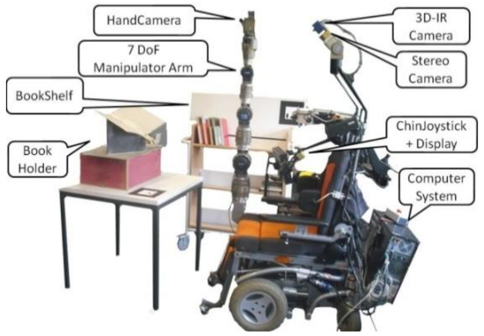

Figure 1: FRIEND-System in library scenario

To the best of our knowledge there exist up to now only few systems for book recognition [1, 2]. Those systems use easy identifiable markers or codes on each book or optical character recognition for book localization on a book shelf in a library. Compared to other approaches, the work presented here does not depend on any identifier label or a priory knowledge of the books in the scene and it is able to detect books of different color and size. That is accomplished by combining two algorithms: the first one uses planar segmentation of the scene with the stereo camera and the second one is based on a visual servoing strategy with a gripper mounted camera.

2 Book detection

Two new algorithms for book detection were developed and implemented to ensure robust grasping operation. The first one is based on the stereo vision and planar segmentation, so-called Book Detection with Stereo Camera (*BwS*) and computes the upper right corner of the book to be grasped in 3D. The second one, so-called Book Detection with Hand Camera (*BwH*) which is based on image feature based visual servoing and has to perform two tasks: the first task is to improve (fine tuning) the output point of the *BwS* and the second one is to determine the orientation of the book with respect to the gripper. A third algorithm, called Black Stripe detection with Hand Camera *BSwH*, is used to compute the intersection point between black stripe and right most book on the shelf.

Since the distance between the stereo camera and the book cart is about 1.5 m, the computed point resulted from *BwS* could have low accuracy and cannot be used as grasping point directly. On the other hand, the hand camera which is much closer to the books does also not provide in all cases reliable information due to often changing illumination conditions and shadows; moreover, it cannot provide the depth information of the grasping point. Therefore all the available visual information is combined to determine the grasping point with the necessary accuracy and reliability. The combination of the three algorithms is presented in Section 3.

2.1 Book detection with stereo camera

BwS method was developed to overcome the problems of standard image processing operations such as edge detection, line segmentation and parallelogram extraction and to cope with cluttered scene and quite large distance between the stereo camera and the book cart. *BwS* method is based on the fact that each book consists of several planar surfaces, each one representing one side of the book. In the working scenario each book could display up to three planar surfaces, which are the top side, the front side and the right side of the book. The number of surfaces and the size of each depend on the camera position in relation to the books on the cart. **Figure 2** illustrates the idea of the proposed method.

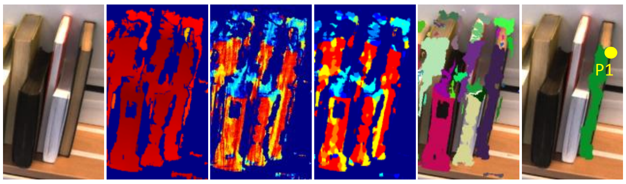

Figure 2: Example of using stereo camera in book segmentation, from left to right: the left stereo camera image (cropped), the corresponding disparity map, the computed *DGoD* image, clustered image, segmented planar regions, re-projected upper right corner *P1* on the right most book.

Using the stereo camera of the robot, the disparity map of the scene is computed using the block matching algorithm [6]. The proposed method segments the disparity into its primitive planar surfaces, where each planar region represents one possible side of a book on the shelf. The segmented regions are analyzed based on their size, 3D plane orientation and location with respect to the book cart in 3D. Candidate objects with 3D plane orientation parallel to the back side of the book cart and perpendicular to the shelf plane are initialized where each candidate object represents the front side of a book on the shelf. The right most book is then extracted from object candidates and the upper right corner *P1* of the extracted region is finally computed in 3D shown in **Figure 7**.

P1 is defined initially in 2D image plane at location (*Ymin*, *Xmax*), where *Ymin* is the minimal y coordinate and x the maximal x coordinate of the extracted region. However, in the case where the location (*Ymin*, *Xmax*) is located outside the extracted region; then the nearest point from the extracted region to the location (*Ymin*, *Xmax*) is considered as *P1*. Validating *P1* is presented in Section 3.

A new approach is developed in this work for planar segmentation, which is based on the Gradient of Depth (*GoD*) feature. Developing new approach has been motivated by the results from other state of art methods which are tailored to segment planar regions from depth images generated from time of flight cameras or laser scanners but failed to segment planar regions from disparity images. **Figure 3** depicts the proposed *GoD*-based planar segmentation algorithm. The input of the algorithm could be of any type of depth images.

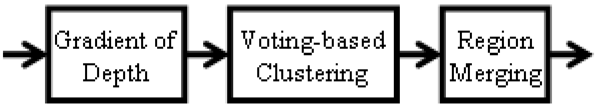

Figure 3: Block diagram of the proposed *GoD*-based planar segmentation algorithm.

The *GoD*-feature at each pixel *p* in the input depth image is defined by two components: Directional Gradient of Depth (*DGoD*) and Magnitude Gradient of Depth (*MGoD*). The *MGoD* and *DGoD* components for pixel *p* are computed using the following equations:

$$DGoD_{p(y,x)} = \arctan\left(\frac{\mathrm{dy}}{\mathrm{dx}}\right) = \arctan\left(\frac{P_{(y+1,x)}-P_{(y-1,x)}}{P_{(y,x+1)}-P_{(y,x-1)}}\right) \qquad (1)$$

$$MGoD_{p(y,x)} = \sqrt{\left(P_{(y+1,x)} - P_{(y-1,x)}\right)^2 + \left(P_{(y,x+1)} - P_{(y,x-1)}\right)^2} \qquad (2)$$

where $P_{(y,x)}$ is the depth value at the image coordinate (y,x). **Figure 4** illustrates the idea of *MGoD* and *DGoD* components on a synthetic range image. The range image contains two boxes, one partially occluding the other, on the left, cylindrical object in the middle and sphere object on the right. As it could be seen; pixels belong to the same planar surface or parallel surfaces have the same *DGoD* value while pixels belong to surfaces with different orientations have distinct *DGoD* values. On the other hand, pixels with high *MGoD* values refer to jumping edges in the depth levels and the *MGoD* image is used later in the clustering process. The output of (1) is in the range of [0° 360°]. There is a special case in computing (1) that should be taken into consideration: 0° and 360° refer to two different planes: in the case of 0° the considered region is parallel to the image plane (see the middle region of sphere in **Figure 4**) whereas 360° refers to a planar region that has gradient in X direction and the gradient along the Y direction is equal to zero (i.e. *dy = 0* and *dx > 0*, see the right most region of the sphere).

However, due to noise in disparity images; pixels belong to the same surface may have small differences in the computed *DGoD* value, see **Figure 5**. To overcome that

problem, a voting-based clustering process is proposed. The clustering process is based on a voting histogram (known as orientation histogram in other works [4]).
For each pixel in the computed *DGoD* image, 1D orientation histogram with a predefined cluster is initialized for (n × n) neighborhood region. The X axis of the histogram is the clusters bins and the Y axis is the number of votes per cluster. Each pixel in the neighborhood region votes for a specific cluster in the histogram and the pixel value in the output clustered image belongs to the cluster with the highest number of votes.

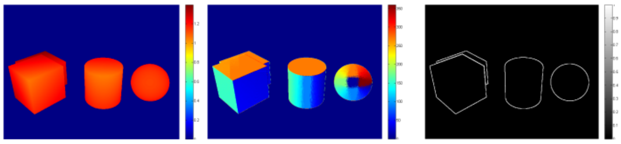

Figure 4: Synthetic range image (left), *DGoD* image (middle) and *MGoD* image (thresholded for clarity) (right).

To prevent merging two parallel adjacent planes into one cluster (like the case of the upper side of the boxes in **Figure 4**), pixels with high *MGoD* value are removed from the clustered image. The output clustered images contains disjoint clusters where each of them is considered as initial segmented region. A merging process is then applied on the initially segmented image. In the merging process; two adjacent regions are merged if the 3D distance between the point of one region and the plane of the other region is smaller than a predefined threshold T_{dis} (T_{dis} = 0.75 in this work). This process is repeated until no more regions are merged. Table 1 shows the performance evaluation of the *GoD* based method on Perceptron Test Set [11] compared to other methods. The evaluation methodology used in this work is described in [3].

Method	GoD	EG	UE	UB	USF	WSU
Correct seg. regions	73%	72%	68%	65%	60%	40%

Table 1: Evaluation of different segmentation methods on Perceptron Test Set at threshold tolerance of 0.8 [3]. EG: edge detection based on scan line approximation, UE: Gaussian and mean curvature clustering based on Local Surface Normals (LSN), UB: region growing based on scan lines, USF: region growing based on LSN, WSU: Principal components clustering based on LSN.

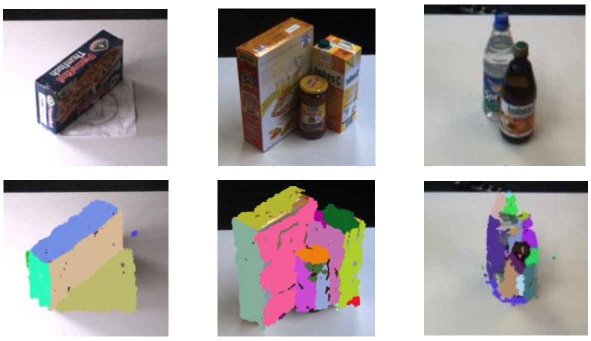

Figure 5: Segmented regions from nonplanar objects.

In addition to the high segmentation accuracy of the GoD segmentation, the proposed algorithm is able to segment planar regions on nonplanar objects in a cluttered scene as well, see **Figure 5.**

2.2 Book detection with hand camera

The robot arm moves to (x, y) coordinates of the point *P1* with a fixed z coordinate in front of the book cart in gripper coordinate system (see **Figure 7**). After reaching the target point, hand camera is initialized. The image from the hand camera is pre-processed to filter out noise and it is transformed to an edge image via canny algorithm. Books appear as regions in the edge image, and are separate by its contour from other books (**Figure 6a**). The use of a contour detection algorithm yields a set of possible book candidates (**Figure 6b**). Candidates are analyzed based on different selection criteria like geometrical shape, size and the location to black stripe. One challenge here comes from shadows on the book front side. The consequence is that one book can be split into several candidates by the contour detection algorithm. Anything written or stamped on the book pages have the same effect. Book candidates are post-processed and candidates which belong to one book are merged. This is done by comparing the boundary edges of each candidate in terms of its location and slope. An example of the results of the presented algorithm is shown in **Figure 6b**. The right most book in the camera view is then extracted from the book candidates and the upper right corner of the segmented book is computed (point *R* in **Figure 6c**). The robot end effector tracks the point *R* using image feature based visual servoing until it is positioned in the centre of the image and the gripper is aligned with the book slope. The output of *BwH* is the position of the robot end-effector in 3D (point *P2* in **Figure** 7).

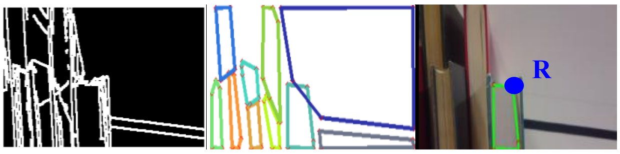

Figure 6: Example for the main steps of the book detection algorithm via hand camera: a) pre-processed image, b) detected book candidates, c) original image with detection of right most book.

2.3 Black stripe detection with hand camera

The horizontal black stripe marker on the vertical board of the book cart is used by *BwS* algorithm as an additional measurement for the 3D verification process of the resulted point *P1* as described in Section 3. The stripe can be detected with a success rate of 100% even under bad lightening conditions, which yields a robust feature. The output of *BSwH* is the 3D point (*PS*) which refers to the point which results from the intersection between the black stripe and the right most book as shown in **Figure 7**. The manipulator starts at position *A* in front of the book cart marker (**Figure 7**). Via image processing techniques as threshold segmentation and canny edge detection the black stripe is tracked by the hand camera using position based visual servoing [7] until the end of the black stripe is in the centre of the image (position *B* in **Figure 7**) and the right most book is supposed to be visible inside the image. The position of the robot end effector defines the x and y coordinates of the output point *PS* and the z coordinate is pre-defined in front of the book cart as shown in **Figure 7**.

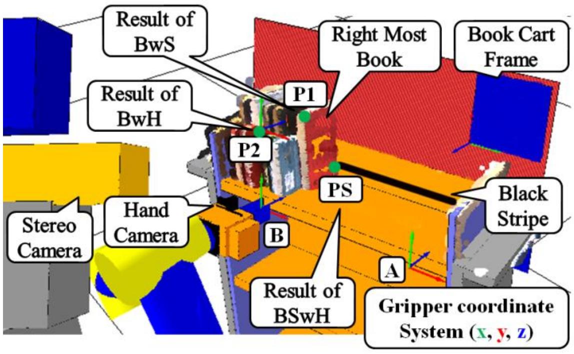

Figure 7: Stations of visual servoing approach.

3 Combination strategy

Figure 8 shows the proposed combination strategy which implements two types of validation processes: the validation of *P1* resulted from *BwS* which in turn used as an input for the second validation process of point *P2* resulted from *BwH*. The output of the combination strategy is the grasping point *GP* used by the manipulator end-effector. In order to validate point *P1* the external measurement *PS* computed independently by *BSwH* is used. This measurment represents the possible location of the next book to be grasped in Y direction of the gripper coordinate system. When a new book cart is delivered for the catalogization process, the point *PS* in the first run is computed from *BSwH* algorithm as described in Section 2.3. After a successful grasping operation the point *PS* for the next run is updated without the need to run *BSwH* algorithm again by using the following equation: *PS.y = GP.y – BW*, where *GP* and *BW* are the grasping point and the book width from the previous run consequently. The book width is computed directly from the distance between the gripper plates after a successful grasping operation.

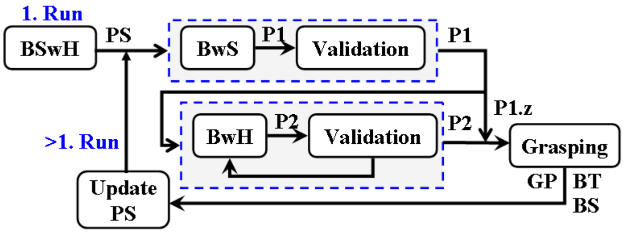

Figure 8: Combination strategy between *BwS* and *BwH*.

3.1 Validation the output of *BwS* algorithm

Since the point *P1* represents the upper right corner of the right most book, it is considered as a valid point if the distance between *P1* and *PS* in the y direction is equal to zero (or within few millimetres of tolerance). As experiments show, the previous condition is not fulfilled in the cases where the book thickness is around 1cm and the segmentation algorithm fails to extracts the front side of the book. If *P1* is not valid, the result of *BwS* algorithm will be ignored and *BwH* algorithm tracks the upper right corner as described in Section 3.2 starting from the position defined by the point *PS* instead of *P1*.

3.2 Validation the output of *BwH* algorithm

According to Section 2.2 *BwH* extracts the right most book candidate and the y coordinate of *P2* is compared to the y coordinate of *P1* as follows:

$$\begin{aligned}&if\ \left|P1_y - P2_y\right| && < \ tol,\ then\ P2_y\ is\ valid\\ &else\ if\ (P2_y - P1_y) && < \ 0,\ then\ P2_y\ is\ valid\\ &else\ \ select\ next\ region..\end{aligned}$$

In other words, if the absolute distance in Y-direction is less than the grasping tolerance (1.5 cm, see Section 4.1), then *P2*.x is valid. Otherwise two cases must be distinguished. In the first case when *P1* is located on the right of *P2* in Y-direction, then the result *P2.y* from *BwH* is chosen, since *BwH* has already delivered the right most region and no other candidate region is available on the right. In the second case where *P1* is located on the left of *P2*, that means *BwH* has chosen false region and the next region from the right is chosen as book candidate.

Similarly, validating the x coordinate of *P2* is expressed as follows:

$$\begin{aligned}&if\ \left|P1_x - P2_x\right| && < \ tol,\ then\ P2_x\ is\ valid;.\\ &else\ if\ (P2_x - P1_x) && > \ 0,\ then\ P2_x\ is\ valid\\ &else\ \ P2_x = P1_x.\end{aligned}$$

In the case where the absolute difference between *P1* and *P2* in X direction is bigger than the grasping tolerance (2 cm, see Section 4.1) and *P1* is located above *P2* in X direction, then *P2.x = P1.x* is chosen. This is based on the fact that *P1* is computed from the point cloud of the scene and the position of *P1* cannot be higher than the real upper book corner.

4 Motion planning and grasping

The position of the screen of the user and the position of the book cart imply the possibility of contact between these elements and the robot arm. Therefore motion planning is necessary to avoid any of these contacts. The algorithm calculates a collision free trajectory for the robot arm (**Figure 9b**). The motion planning algorithm, used in this paper, is called *CellBiRRT* [8, 9]. It is based on Rapidly exploring Random Trees (RRT). The main advantage is the ability to solve fast (average time below 2 seconds

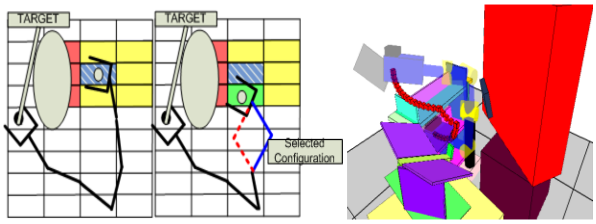

Figure 9: a) Cell Division-Selection, b) Trajectory.

for this setup) path planning problems while at the same time it can handle extra position and orientation constraints of the end-effector. That is done by dividing the Cartesian space into cells and selecting an appropriate one as a basis for generating random configurations.Figure 9a presents the cells and Figure 9b a trajectory result for the placement on book cart scenario. For further reading refer to [8]. For collision detection the algorithm described in [9] is used.

4.1 Tolerance for book grasping

In this section the allowable sizes of books based on the results from *BwS* and *BwH* algorithms is presented. The book thickness can be calculated from the extracted image information by the following formula:

$$bm = 2\tan(\alpha)\cdot(d1+d2)\cdot bp\cdot dp^{-1} \qquad (3)$$

The context and the meaning of the used parameters are shown in **Figure 10**. The distance *d2* is calculated via *BwS* and *bp* via *BwH* algorithm. All other used parameters are constant.

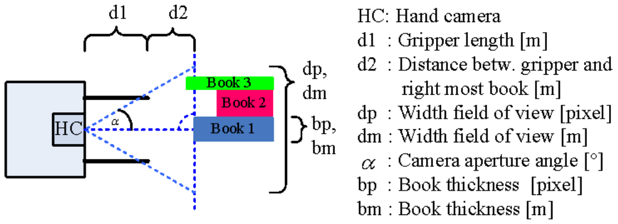
d1
d2
Book 3
Book 2
Book 1
HC
α
dp, dm
bp, bm
HC: Hand camera
d1 : Gripper length [m]
d2 : Distance betw. gripper and right most book [m]
dp : Width field of view [pixel]
dm : Width field of view [m]
α : Camera aperture angle [°]
bp : Book thickness [pixel]
bm : Book thickness [m]

Figure 10: Calculation of the book thickness.

The experiments show, see Section 5, that the previous formula could not compute the book thickness reliable and as consequence it is not used in the following anymore.

Suppose L_{max} is the maximum opening of the gripper (6 cm) . The end effector (the middle point of the gripper) moves to the grasping point *GP= (P2.x, P2.y, P1.z)*. Since the book size is unknown, the gripper opens fully and the maximum book width allowed to be grasped in that case is defined as $S_{max} = L_{max}/2$ (**Figure 11a**). On the other hand, to avoid the collision with book *B3* (**Figure 11b**), a minimum width of two sequential books should satisfy the following condition $W_{min} \geq S_{max}$ where $W_{min} = W_{B1} + W_{B2}$, $W_{B1}$and W_{B2} are the widths of books *B1* and *B2* respectively. If the minimum size of book is defined as $S_{min} = S_{max}/2$, the sum of width of two sequential books will be equal to S_{max} and avoids the collision with book *B3* (**Figure 11c**). Accordingly, the grasping tolerance in Y direction is equal to 1.5 cm.

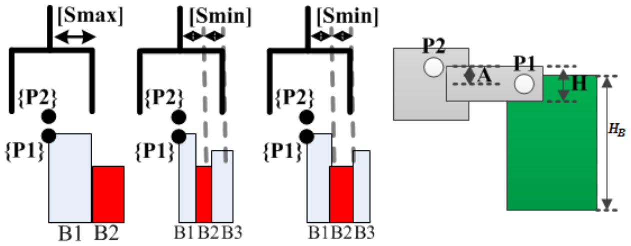

Figure 11: Tolerances for grasping a book: a-c) in Y direction, d) in X direction of gripper coordinate system.

As the robot grasps the book at its upper part, the grasping tolerance in X direction is defined as the half of the plates height (*A=2 cm*) (see **Figure 11d**).

5 Experimental results

In order to evaluate the performance of proposed vision system in this work; four different test sets were carried out. If not mentioned differently the gripper coordinate system is chosen as reference coordinate system. The first test set includes the ability of the robot to successfully grasp a book using the proposed combination strategy. In 20 consecutive runs, the robot was able to grasp 15 books meaning that a success rate of 75% was achieved. The analysis of the failed cases show that in two cases the image processing algorithms were not able to deliver accurate grasping points and the rest of the failed cases comes from errors in calibrating the robot with respect to its scene, leading to a collision of the gripper with other books. The rest of the test sets are aimed to evaluate the accuracy of the proposed image processing algorithms described in this work.

The graph in **Figure 12** represents performance evaluation of *BwS* algorithm. The graph shows the absolute difference in gripper coordinate system between the computed upper right corner *P1* of the book to be grasped and its corresponding ground truth point which has been manually selected. From 20 consecutive runs, only two cases have failed to deliver the correct point whereas in all other cases the average error is 0.0052 m, 0.0117 m and 0.0038 m for the X, Y and Z direction respectively. The evaluation process shows that *BwS* tends to produce accurate results in both X and Z direction. The average error in Y direction (0.0117 m) is smaller than the grasping tolerance defined in Section 4.1 (0.015 m), however, in four cases (run 5, 6, 13, 20) the difference is biggerimg1, which is caused by errors in computing the disparity image resulted from the low texture books. This justifies the using of *BwH* algorithm as an assistant algorithm to increase the grasping reliability.

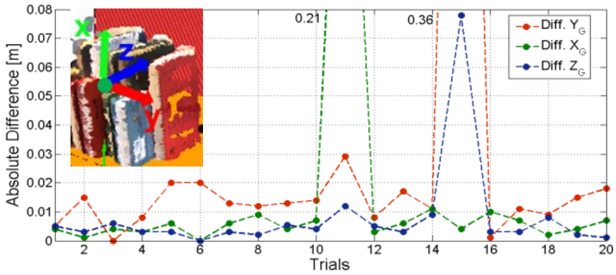

Figure 12: The absolute difference in gripper coordinate system between the computed point *P1* and its corresponding ground truth point, the small figure (left above) shows the reference gripper coordinate system.

Another 20 consecutive runs shown in **Figure 13** were performed which shows the difference between *P1* and *P2* resulted from *BwS* and *BwH* respectively. A successful grasping point is guaranteed when the difference between points *P1* and *P2* in gripper coordinate system is smaller than 0.015 m in Y direction (the blue plot in **Figure 13**) and smaller than 0.02 m in X direction (the red plot in **Figure 13**) as described in Section 4. As evident, 80% of the test cases produce accurate grasping points. The big differences in runs 10 and 15 are caused by the *BwH* which has missed the upper book edge due to shadows. In run 5 *BwS* failed to detect the upper right corner.

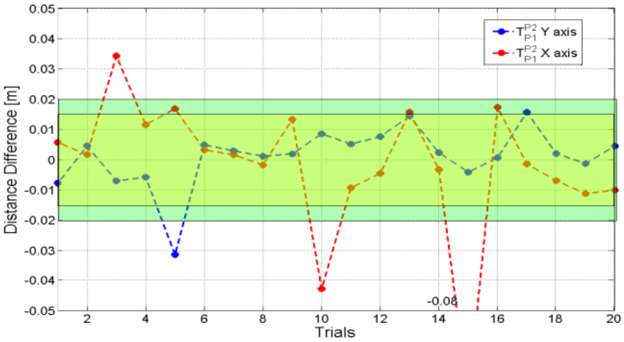

Figure 13: Difference between points *P1* and *P2* for 20 consecutive runs, the yellow and green bars are the grasping tolerance in Y direction and X direction respectively.

Figure 14 shows the achieved detection accuracy of the book thickness from the BwH algorithm. As it can be seen the computed book thickness (the width of the book) after (3) is not stable. The deviation is less than 0.01 m in only 65% of the tested cases. The reasons come from inaccuracies by the calibration procedure, the determination of the current gripper position and by the calculation of the book thickness bp in the image plane from the hand camera (in pixel). On the other hand experiments have shown that the detected book slope is every time in the range of +-3°, which is sufficient for successful grasping.

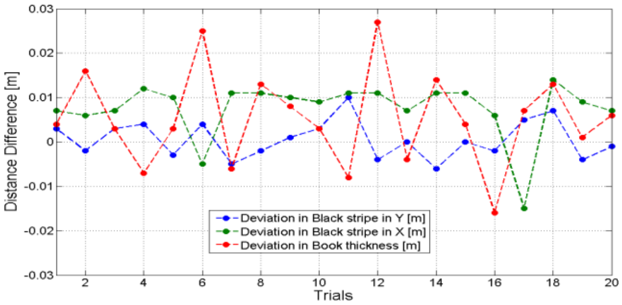

Figure 14: Achieved detection accuracy of book thickness and end of black stripe detection.

Additionally **Figure 14** shows, that the detection of the end of the black stripe is very accurate and every time inside the chosen positioning accuracy in the *BSwH* algorithm, 0.01 m in Y direction and 0.015 m in X direction.

6 Conclusion

In this paper two new book detection algorithms were combined in order to achieve reliable book detection and grasping in library scenario. The evaluation experiments showed that the proposed combination of two different book detection algorithms increases the reliability of the book detection and grasping. The presented approach will be improved in future. Filter algorithm for shadow removal will be integrated to increase the robustness of the proposed method even under worse lightening conditions. The calculation of the book thickness will be improved in order to achieve higher reliability. The second right most book will also be detected and used together with the book thickness to avoid collision of the gripper with other books.

7 Acknowledgment

The authors would like to thank the Integration Agency Bremen which is part of Pension Office in Bremen, assigned to the Department of the Bremen Senator for Profession, Women, Health, Youth and Social Affairs, which funds the project 'ReIntegraRob' and the State and University Library of Bremen (SuUB) with which the project is performed in co-operation.

8 Literature

[1] A. Leonardi, S. Messelodi and L. Stringa, "Book recognition as an example of flat and structured objects classification", in Cybernetics and Systems, 1995, pp. 621-645

[2] M. Prats, E. Martinez, P. J. Sanz and A. P. del Pobil: The UJI librarian robot, Intel Service Robotics, 2008, pp. 321-335

[3] A. Hoover et al: "An experimental comparison of range image segmentation algorithms", IEEE Transactions on Pattern analysis and machine intelligence, vol. 18, no. 7, 1996.

[4] F. Alhwarin, D. R.–Durrant, and A. Gräser, "VF-SIFT: Very Fast SIFT feature matching, " Pattern Recognition, Lecture Notes in Computer Science, vol. 6376, pp. 222-231, 2010

[5] X. Jiang and H. Bunke, "Edge Detection in Range Images based on scan line approximation" Computer vision and Image understanding, vol. 73, no. 2, 1999.

[6] T. Tao, J. C. Koo and H. T. Choi: "A fast block matching algorithm for stereo correspondence", IEEE conference on Cybernetics and Intelligent Systems, 2008

[7] S. Hutchinson, G. D. Hager and P. I. Corke: "A tutorial on visual servo control", IEEE Transactions on robotics and automation, vol. 12, no. 5, 1996, pp. 651-670

[8] C. Fragkopoulos and A. Gräser: "A RRT based path planning algorithm for Rehabilitation robots"; ISR / Robotics, Munic; 2010

[9] C. Fragkopoulos and A. Gräser: "Dynamic efficient collision checking method of robot arm paths in configuration space", IEEE/ASME International Conference on Advance Intelligent Mechatronics, 2011

[10] ReIntegraRob-Webpage: http://www.iat.uni-bremen.de/sixcms/detail.php?id=1268

[11] Perceptron range images database: http://marathon.csee.usf.edu/range/DataBase.html