Calibration of RGB Camera With Velodyne LiDAR

Martin Velas, Michal Spanel, Zdenek Materna, Adam Herout
Department of Computer Graphics and Multimedia
Faculty of Information Technology, Brno University of Technology
Bozetechova 2
612 00, Brno, Czech Republic
{ivelas, spanel, imaterna, herout}@fit.vutbr.cz

ABSTRACT

Calibration of the LiDAR sensor with RGB camera finds its usage in many application fields from enhancing image classification to the environment perception and mapping. This paper presents a pipeline for mutual pose and orientation estimation of the mentioned sensors using a coarse to fine approach. Previously published methods use multiple views of a known chessboard marker for computing the calibration parameters, or they are limited to the calibration of the sensors with a small mutual displacement only. Our approach presents a novel 3D marker for coarse calibration which can be robustly detected in both the camera image and the LiDAR scan. It also requires only a single pair of camera-LiDAR frames for estimating large sensors displacement. Consequent refinement step searches for more accurate calibration in small subspace of calibration parameters. The paper also presents a novel way for evaluation of the calibration precision using projection error.

Keywords

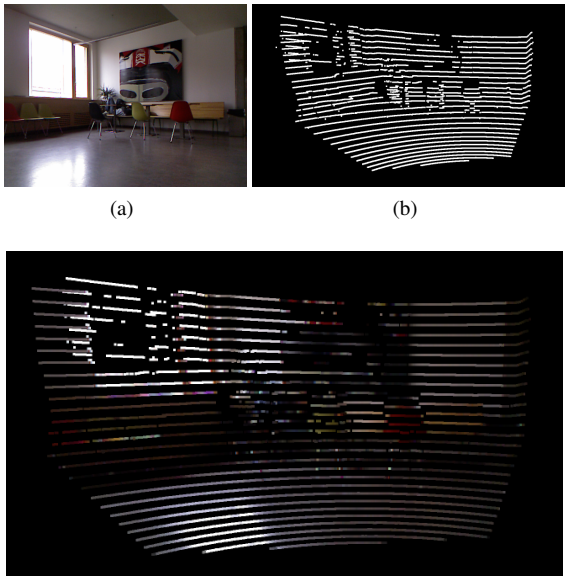
camera, LiDAR, Velodyne, calibration, marker

1 INTRODUCTION

This paper deals with an automatic calibration of RGB camera with Velodyne LiDAR (Light Detection And Ranging) sensor (also called laser radar or scanner). The Velodyne sensor scans the whole area around the rotating beam using 32 or 64 laser rays and finds its use in many applications such as autonomously driving Google car. Using techniques proposed in the following chapters, the extrinsic parameters – position and orientation of the LiDAR related to the camera – can be computed as is shown on Figure 2. This problem is commonly named as searching for 6 *degrees of freedom (6DoF)* - rotation and translation against all the three axes of 3D space.

When solving such calibration problem, the most challenging tasks are: the ability to deal with large sensor displacement and minimal requirements for a specific scene setup while preserving accuracy of the resulting calibration.

In order to fulfill these goals, the proposed camera-LiDAR calibration pipeline performs the calibration in two consequent steps. First, a large sensor displacement is estimated using a special, but still simple, 3D marker and then a refinement is applied in order to obtain a more accurate calibration. Using the proposed projection error, it will be proved that the coarse calibration reaches similar or even better accuracy than the manual one (the operator sets and visually verifies the



(a) (b)

(c)

Figure 1: Using the calibration for fusion of the camera image (a) and Velodyne scan (b) obtaining a coloured point cloud (c).

 ISBN 978-80-86943-71-8

calibration parameters) in the task of sensor displacement computation. The consequent refinement process further improves the calibration accuracy and exhibits 5% better results than the previously proposed method by Levinson and Thrun [LT13].

Fusion of the aligned camera with the LiDAR sensor was recently used in many tasks of computer vision in order to enhance their performance. Douillard et al. [DBR09] use a 3D classifier based on the fused Velodyne data with the monocular color imagery for modelling semantic content in scenes. Colour point clouds obtained through the camera-LiDAR fusion proved to be useful in velocity estimation for precise vehicle tracking and autonomous driving [HLT13]. Figure 1 demonstrates results of the Velodyne point cloud colouring. A combination of the aligned LiDAR with the camera was also successfully used in the process of building 3D geological maps [NMV10] or for preserving cultural heritage by the 3D modelling of historical buildings [BJCK11].

The method described in this work extends the solution proposed by Levinson and Thrun [LT13]. The biggest asset of the mentioned paper is the proved assumption that the edges can be robustly detected in both the camera image and the LiDAR scan.

The novelty of our method is the two-level calibration scheme dividing the problem into two consequent parts. The large displacements in sensors position are computed using the camera-LiDAR correspondences found via a fully automatic 3D marker detection. The novel marker also allows to perform the coarse calibration from a single camera image and the corresponding LiDAR scan what is an advantage over traditional "multi-frame" techniques. Finally, the obtained calibration is refined using a simple exhausting dense search in a small subspace of the calibration parameters. Both the marker detection and the refinement are based on the robust edge detection proposed by Levinson and Thrun [LT13].

In its experimental part, the paper also introduces a new way how the calibration methods can be evaluated automatically using an image segmentation. The process assumes that in the calibration scene foreground and the background can be easily separated both in the camera image and the LiDAR point cloud. As the error metric, the ratio of the correctly projected 3D points onto proper image segments is used.

Using this metric will be proved that our modification of the refinement process reaches 4% lower miscalibration error than the basic solution.

2 RELATED WORK

Methods of the automatic camera calibration with the LiDAR sensor can be divided into several groups. The first group of methods [KP10, PMSE10, GMCS12] requires a chessboard-like marker for an automatic calibration. Other methods [TN13, LT13] assume an arbitrary markerless scene for automatic calibration but their capability of calibration is reduced just to a refinement of small differences in position and orientation between sensors. Some of the recently used methods [BCK13, LN09, PPRJ13] use a sequence of captured frames and compute the calibration parameters from the motion obtained through the feature tracking or special IMU units.

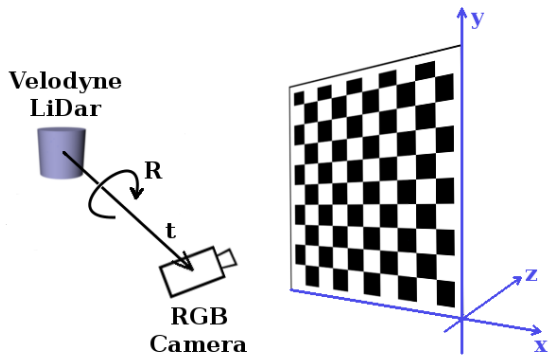


Figure 2: Problem of a camera – Velodyne calibration. The solution of the problem are the vectors t and R describing the translation and the rotation of Velodyne LiDAR related to the RGB camera.

An automatic alignment of the camera with 2D laser rangefinder where typical checkerboard marker must be observed from multiple views was proposed by Zhang and Pless [ZP04]. Constraints based on this observations are solved minimizing an algebraic error and followed by a nonlinear process. The external calibration of a 3D laser scanner with an omnidirectional camera system [PMSE10] requires also a planar checkerboard pattern to be observed simultaneously from the laser scanner and the camera system from a minimum of 3 views. Normals of the planar surface form a non-linear optimization problem that is solved for the extrinsic calibration parameters. All the mentioned techniques suffer from the necessity of the presence of some known marker in the scene. Moreover the marker must be observed from multiple points of view. A toolbox for the automatic camera-Velodyne calibration presented by Geiger et al. [GMCS12] recovers intrinsic and extrinsic camera parameters using a single view of multiple chessboard instances placed in the scene what is still quite impractical.

A structure-from-motion technique [BCK13] for camera-laser fusion which uses a 2D laser sensor and multiple cameras, was also presented. The laser points are projected onto the images and tracked using KLT algorithm [ST94] to other frames in order to be used as 3D-2D correspondences. The paper by Núñez et al.

[PPRJ13] proposes a method to perform the extrinsic calibration between a camera and a 3D-LRF observing a checkerboard pattern with the aid of motion obtained by Inertial Measurement Unit (IMU). The limitation of these methods is the need for multiple instances and this special type of hardware.

A “golden” calibration method which would require no marker, single observation of the scene, no further HW requirements and still provides sufficient results is still missing. Recently, techniques which work with markerless arbitrary scenes assuming some shared property by both the camera image and the LiDAR scan appears. Method [TN13] is based on a hypothesis that the horizontally oriented planes are displayed on the camera image with higher intensity. The searching for the proper calibration parameters is performed as a particle swarm optimization [KE95] with the cost function based on mutual information between the image and projected point cloud also used by Pandey et al. [PMSE12]. However, this techniques come with an other limitation which is the need for a small displacement of the sensors in both their orientation and position.

A real time miscalibration detection and the transformation adjustment introduced by Levinson and Thrun [LT13] enables online and automatic camera-laser calibration in arbitrary environments. In the mentioned work, the miscalibration detection is performed by probabilistic monitoring. Consequently, a continuous calibration optimizer adjusts transform offsets if miscalibration was previously detected. This solution is based on the assumption that edges can be robustly and reliably detected in both the camera image and the LiDAR scan which proved to be right. The objective function based on the matching edges detected in both modalities and which is used for the optimization is convex only in a small subspace around the global optimum. Thus the calibration task is limited to only small variations in the position and the orientation between the camera and LiDAR.

The main drawback of the two previously mentioned methods [LT13, TN13] – the limitation of the position and the orientation between sensors just to a small subspace of 6DoF parameter space – is relaxed in this paper using a special type of 3D marker which allows coarse to fine calibration of the sensors using just a single frame captured by the camera and LiDAR from a single point of view and a single instance of the marker in the scene. Our method of marker detection is based on the proved assumption [LT13] that edges can be robustly detected in both the camera image and the LiDAR point cloud. The final step of the sensor alignment proposed in this paper is the calibration refinement by means of searching in a small subspace of calibration parameters.

3 CALIBRATION OF CAMERA WITH LIDAR

Our proposed solution consists of two consequent steps. First, an approximate coarse calibration is estimated. In this step, we assume that the translation of the laser sensor against the camera is much more significant than the rotation. For the coarse calibration step, a special "3D marker" which can be easily found in both the camera and the Velodyne data was designed. Found correspondences are further used for deterministic computation of the translation between the camera and Velodyne.

In the next step, the calibration of sensors including the rotation is refined.. This fine calibration is based on the solution proposed by Levinson and Thrun [LT13] with further simplifications which brings computation speed-up and higher reliability of obtained results.

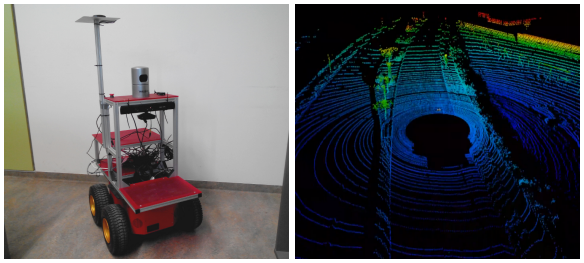
To be more specific, the coarse to fine calibration pipeline works in the following steps:

1. *Coarse calibration*
   (a) edge detection in the Velodyne point cloud and the camera image,
   (b) 3D marker detection (circles’ centres and the radius),
      i. in the camera image, using the Hough transform [YPIK90],
      ii. in the point cloud, using our detection algorithm based on *RANSAC* [FB81] which is described below in detail.
   (c) Estimation of the translation between both the sensors in the 3D space.
2. *Calibration refinement*
   (a) initialization using estimated coarse calibration parameters,
   (b) dense search in the small subspace of the calibration parameters,
      i. projection of the point cloud with detected edges to the image plane
      ii. edge detection and the *inverse distance transform* of the camera image
      iii. cross correlation of these two images
   (c) pick the calibration solution based on the mentioned cross correlation criteria

Sensors Description

Calibration techniques presented in this work were developed and tested with sensors mounted on our robotic platform shown on Figure 3a.

 ISBN 978-80-86943-71-8



(a) (b)

Figure 3: Our robotic platform (a) and an example of the Velodyne scan (b).

The source of the camera image is the Kinect© [1] sensor mounted on our robot. Besides the RGB frames it provides a depth map which is not used in the calibration process.

Source of the laser radar scan is the Velodyne© [2] LiDAR sensor – the HDL-32 model with 32 lasers having field of view 360° horizontally and 40° vertically. The sensor rotates 10 times per second and captures 700 000 points per second which are organized into 32 rings (one per each laser). An example of the scan provided by this sensor is displayed on Figure 3b.

The robot runs ROS Hydro for unificated control and the access to sensoric data. We assume that the sensors themselves are already calibrated and the ROS topic interfaces provides auxiliary data as the intrinsic camera parameters (focal length and the principal point) in form of the projection matrix.

Coarse Calibration

The initial step of the calibration estimates the coarse calibration parameters using the novel 3D marker. This estimation allows a large position displacement of the sensors and it is further refined in the consequent step.

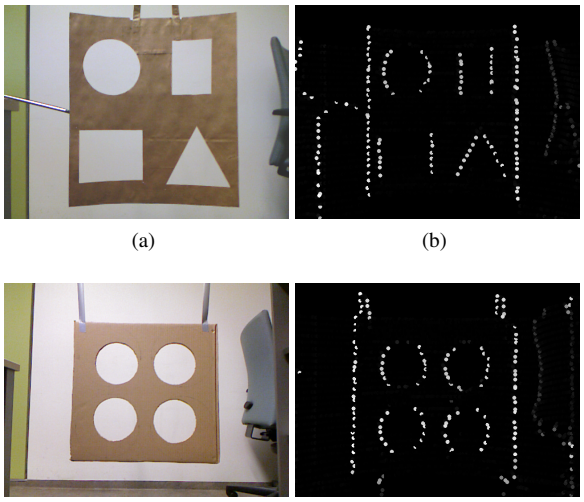
Marker Description

Usage of a marker for the calibration purposes is nothing new. A typical example of marker used for calibration is chessboard pattern which was also successfully used for the camera-LiDAR calibration [GMCS12, PMSE10, KP10]. The biggest drawback of this marker is the need for multiple observations of the marker from different view points.

The design of our marker undergoes the condition that it has to be easily detected in both the camera image and the Velodyne data and only one observation of the marker is sufficient for estimation of the calibration parameters. The pre-processing of the camera and the LiDAR data includes an edge detection which will be described in the next sections.

[1] http://www.xbox.com/en-US/kinect
[2] http://velodyneLiDAR.com/

Figure 4b shows how edges of various geometrical shapes are captured using the Velodyne LiDAR. One can observe that the vertical position of shapes with horizontally oriented edges can not be determined precisely with Velodyne-like sensor which scans the surrounding environment in horizontal rings.



(a) (b)

(c) (d)

Figure 4: Planar marker with holes of different geometrical shapes (a) and edges detected in its LiDAR scan (b). Our proposed marker (c) and the detected edges (d).

Figure 4d shows the final version of our plannar marker with circular holes in it. Thus, both the horizontal and the vertical position of the circles can be clearly determined visually and automatically.

Detection in Velodyne Data

Edge detection in Velodyne data is based on the assumption that depth discontinuities in laser data represent edges in the real scene [LT13, BCES11].

The frame of Velodyne LiDAR is processed through the rings as it is natural way for this type of the sensor. It is possible to assign a value to each measured point which depends on its depth measurement and the depth of its direct neighbours:

$$X_i = max(P^r_{i-1} - P^r_i, P^r_i - P^r_{i+1}, 0)^\gamma \qquad (1)$$

where P^r_i represents the range of the i-th point of the point-clout gained from Velodyne and γ is the constant ($\gamma = 0.5$ was used in our experiments). The lower γ is, the higher value will be assigned to the points of low distortion relatively to the points with high distortion. Afterwards, the X_i values are normalized into the range $(0;1)$.

For efficiency, the points with assigned value under a certain threshold (0.1) are discarded. This leads

 ISBN 978-80-86943-71-8

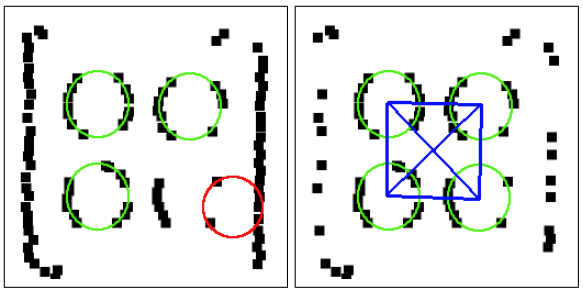
to reduction of the number of points (about 90% of points are removed) while preserving the detection accuracy. The threshold value was derived from the work of Levinson and Thrun [LT13]. Our experiments confirmed that the suggested value is reasonably restrictive (values 0.01 or 0.03 might result in a loss of significant information while higher values 0.3 or 0.5 do not bring any detection improvements and slow down the computation).

For the detection itself we assume that the size, number and mutual position of circles in the marker are known. Another assumption is that the marker is visible for both Velodyne and the camera and the circular holes in the marker can be considered as a planar object - an intersection of a plane with spheres. These assumptions are used for the detection verification.

The marker detection in the Velodyne point cloud repeats three major phases - detection, verification and pointcloud pruning:

1. The marker is roughly located as the plane detected by the *RANSAC* algorithm [FB81] with the model of the plane. The outliers are discarded and the rest of the points are further processed.

2. Borders of the marker are removed using the RANSAC algorithm for the line detection in 3D space. These borders would cause inaccuracies in the circles detection process because the Velodyne data are sparse and only few $(15 - 20)$ points represent each circular hole. Figure 5a shows a situation when the borders were not discarded and the detection failed.



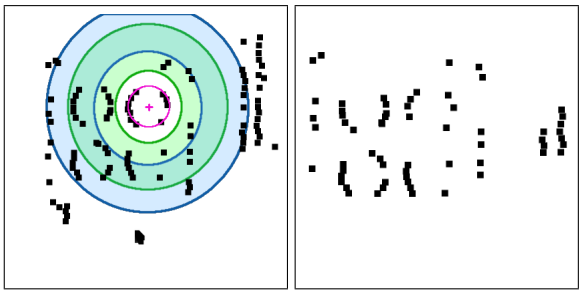
(a) (b)

Figure 5: Situation when the circular holes were found incorrectly (a) because of lack of the data and preserving the vertical borders of the marker. A correct detection of the 4 marker holes after the marker borders are discarded (b). Verification was done by checking distances of circle centres.

3. The algorithm detects 4 best candidates for the marker holes also using RANSAC with model of the sphere (intersection of the spheres and the plane are the circles the algorithm is looking for).

4. After detection of a given number of spheres their mutual position is verified – distances of the centers (blue lines on Figure 5b) are checked. If verification succeeds the whole marker is detected and the algorithm ends.

5. Otherwise the algorithm continues with a phase of pointcloud pruning for the next iteration of the algorithm. The input of this step is the same set of points representing the planar marker (see Step 1). As we know the mutual distance of the circle centres in the marker, we can say for the each detection where the 3D points of the other circles should be located – coloured areas on Figure 6a. The algorithm preserves only the points which belong to such an area of at least one the detections found in Step 3. If at least one of the detections was correct, this step of the algorithm preserves all of the points belonging to the circle holes of the marker. The algorithm then continues with Step 3.



(a) (b)

Figure 6: Process of the pointcloud pruning (a). Each detection (purple circle) yields the coloured areas (green for nearer circles and the blue for the circle laying on diagonal) where the points of other marker circles may be located. These points are preserved and process is repeated for each detection. Remaining points are discarded. The detection continues from Step 3 using the resulting pruned point cloud (b).

The progress of the described algorithm can be seen on Figure 7 where the marker was found after two iterations.

Detection in the Camera Image

The detection process of the circles in RGB camera image starts with the edge detection using Sobel operator which produces less noisy outputs then the previously proposed detector by Levinson and Thrun [LT13].

The circles are then detected using Hough transform for circles [YPIK90]. This detection proved to be robust enough and no further validation is necessary.

Translation computing

In the most general case of the calibration, all six degrees of freedom describing translation and rotation

 ISBN 978-80-86943-71-8

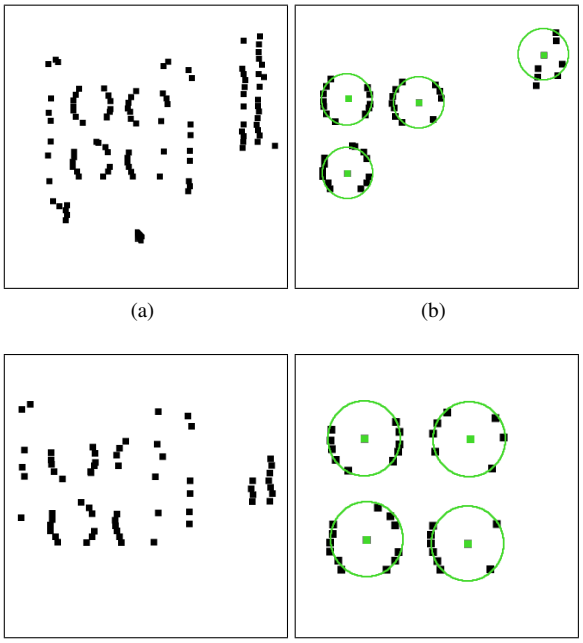


Figure 7: Progress of the marker detection in the pointcloud (a). After first detection (b) the validation fails and the new pointcloud is generated (c). In the second iterations the detection succeeds (d).

must be considered. As we assume that the difference of the position between the camera and LiDAR is much more significant than the rotation itself we reduce the potential geometrical transformation just to the translation.

The calibration process transforming homogeneous coordinates of 3D point $[X,Y,Z,1]$ to the point $[\frac{x}{w},\frac{y}{w}]$ in 2D plane can be described using the following equation:

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} = P.C. \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2}$$

where P is the known camera projection matrix and C is the calibration matrix describing geometrical transformation between the sensors. Both the projection P and the calibration matrix C reduced to three *translation* parameters are presented in following equations:

$$P = \begin{bmatrix} f & 0 & o_x & 0 \\ 0 & f & o_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$C = T = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

where f is the camera focal length and $[o_x,o_x]$ are the coordinates of the principal point in pixels – i.e. the known intrinsic camera parameters. The only unknown variables we need to compute are the components of the translation vector $[t_x,t_y,t_z]$. Remember that the rotation between the sensors is omitted in the process of the coarse calibration and these degrees of freedom are computed during the calibration refinement.

The translation vector is estimated from the found correspondences of the circles (centers and radii) gained through the detection of the marker in both the Velodyne scan and the camera image described in previous sections.

The first unknown component of the translation, which is necessary to be solved, is the depth difference between the sensors t_z. Assuming the marker is planar and neglecting the influence of the perspective projection (marker is approximately straight in front of the robot), it is possible to compute t_z just using corresponding radii of circles detected in the image (r_{2D}) and circles detected in the point cloud (r_{3D}). Equation (5) can be derived from Equation (2) considering the mentioned assumptions, where Z is the depth coordinate of the circle center.

$$t_z = \frac{r_{3D}.f}{r_{2D} - Z} \tag{5}$$

Other components of the translation vector can be roughly estimated using Equations (6) and (7) also derived from (2).

$$t_x = \frac{(x-o_x).(Z+t_z)}{f} - X \tag{6}$$

$$t_y = \frac{(y-o_y).(Z+t_z)}{f} - Y \tag{7}$$

Equations (6) and (7) indicate that a single correspondence can be used to estimate the translation parameters.

In this work, the translation is estimated for each circle correspondence independently and then the gained translations are combined using simple averaging.

Fine Calibration

After the approximate coarse calibration is found using the 3D marker, a more precise calibration is estimated by searching for all the 6 degrees of freedom (the translation and the rotation) in a small parameter subspace.

This searching is performed using a dense sampling of the neighbourhood of the approximate calibration vector $[t_x,t_y,t_z,0,0,0]$ obtained in the previous process of the coarse calibration where the rotation was omitted (zero initial rotation parameters in the vector).

 ISBN 978-80-86943-71-8

During the search for the calibration parameters, the ability to evaluate each set of parameters is essential. It means that we need a cost function f (Equation (8)) which assigns a value proportional to the quality of the calibration to each set of 6DoF parameters between the sensors.

$$f : \mathbb{R}^6 \rightarrow \mathbb{R} \tag{8}$$

Cost Function Based on the Edges

We adopt the error metric used by Levinson and Thrun in their online miscalibration detection system [LT13]. Their error metric is based on the assumption that the edges can be robustly found in the camera image and matched with the range discontinuities in the LiDAR point cloud.

Processing of the data during the fine calibration is similar to the processing during the process of the 3D marker detection.

First, the edges are found in the Velodyne point cloud. During this process, value X_i described in Equation (1) is assigned to the each point and it is considered to be intensity value of this pixel. Then the point cloud is transformed using the parameters we want to evaluate. Finally these points are projected on the image plane using the projection matrix (Equation (3)) and can be compared against the preprocessed camera image in 2D space using some traditional similarity criteria.

The camera image preprocessing consists of edge detection – in 2D case using simple Sobel operator creating edge image E. In order to make the cost function smoother, we apply *Inverse Distance Transform (IDT)* [LT13] using L1 norm to the edge image (Equation 9). IDT assigns each pixel on coordinates $[i, j]$ the value $D_{i,j}$ proportional to the distance and the strength of the edge $E_{x,y}$ in the pixel's neighbor on coordinates $[x, y]$.

$$D_{i,j} = \alpha . E_{i,j} + (1-\alpha) . \max_{x,y} \{ E_{x,y} . \beta^{max(|x-i|,|y-j|)} \} \tag{9}$$

The α factor increases strength of the neighbour edges' impact (0.33 was used in our experiments) and the factor β enlarges the area an edge effectively impacts (0.98 was used in our experiments).

The output of the *IDT* applied on image 1a is shown on Figure 8. Application of this transform causes that the closer the parameters of the calibration to the ideal parameters are the higher value of the cost function are reached.

Finally, the cross correlation similarity criteria (Equation (10)) is applied to the projected point cloud I_V and processed camera frame I_C in order to get value of the cost function.

$$S_E = \sum_x \sum_y I_C(x,y) * I_V(x,y) \tag{10}$$

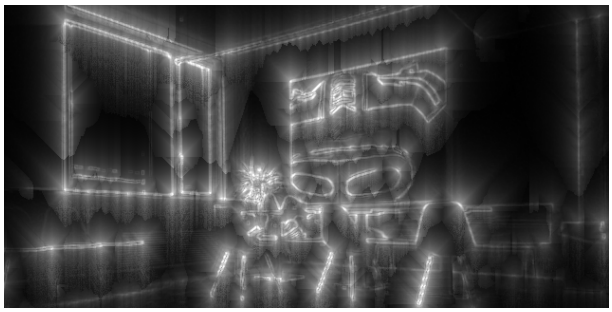


Figure 8: Application of the Sobel operator and *IDT* to the image on Figure 1a

Optimization

Previous works [LT13, TN13] showed that the optimization criteria, such as criteria based on the edge detection, are highly non-convex. It causes that the traditional optimization techniques based on the gradient descent/ascent or Newton's method for the optimization would fail.

In our solution the refinement of the previously established coarse calibration is performed by a regular sampling of the small subspace of the calibration parameters around the point of the coarse calibration. This idea follows the process of regular grid search proposed by Levinson and Thrun [LT13] for their miscalibration detection.

Each point of the calibration parameter space can be evaluated by the criteria (Equation (10)) and this value refers to the fitness of the calibration. The final calibration parameters are than chosen as the point of the parameter space with the maximal cost function or the point obtained by averaging of the multiple points with the fitness value above some threshold.

4 EXPERIMENTAL RESULTS

In our experiments, real data from sensors mounted on the robotic platform (see Figure 3a) were used. The RGB images were captured using the Kinect sensor (43° vertical and 57° horizontal field of view), while the 3D point cloud was scanned using the Velodyne HDL-32E LiDAR producing 32 horizontal scans that cover 360° horizontally and 40° vertically. The depth information provided by Kinect mounted on our robot was used neither for the calibration process nor for the experiments.

Miscalibration Error

A novel criteria to experimentally evaluate quality of the calibration is proposed in this paper. 3D points obtained by the Velodyne LiDAR are projected on the image segment occupied by this object. The projection is performed according to Equation (2) using projection matrix P (known intrinsic camera parameters) and calibration matrix C we try to evaluate.

 ISBN 978-80-86943-71-8

This type of criteria requires a prepared scene so the foreground in both the Velodyne pointcloud and the camera image can be easily distinguished from the background using the the image segmentation. The more 3D points are projected on a wrong image segment, the higher error is obtained.

In this work, a dark 3D marker (Figure 4c) positioned against the white wall was used to evaluate the precision. The segmentation in the camera image was performed using adaptive intensity thresholding. The foreground and the background segment in the Velodyne scan was obtained using *Otsu* thresholding [Ots79] based on the range information of points.

The value of projection error P_E is computed as the number of incorrectly projected points E (i.e. foreground points projected on the background segment and vice versa) divided by the total number of points in the point cloud P:

$$E_P = \frac{E}{P} \tag{11}$$

Coarse Calibration Results

To evaluate the robustness and the accuracy of proposed calibration scheme, we collected data of a distorted 3D marker in front of a wall while adding additional clutter (chairs, shelves, ...) in the background.

Table 1 shows the ratio of correctly detected markers rotated around the X, Y and Z axis to the total number of the detections performed. A marker is correctly detected if the verification phase of detection algorithm succeeded. For each marker pose, 5 pairs of Velodyne and RGB images were retrieved. The algorithm of the marker detection proved to be robust enough to handle these rotations and provides reliable detections for further coarse calibration computation.

	Without verification	Our iterative algorithm
X axis rotation	0.84	0.96
Y axis rotation	0.72	1.00
Z axis rotation	0.76	0.93

Table 1: Percentage of the marker detection (recall) using only RANSAC versus our iterative algorithm with the verification and pointcloud pruning. Marker was rotated around the all 3 axis in range $0-20^\circ$ for this evaluation.

Table 1 shows that the verification and the pruning step of our algorithm brings the significant improvement comparing to the basic approach using only the detection of 4 best candidates for marker circular holes.

Ground truth for the captured data was manually annotated using our tool enabling a user to change the translation between the sensors. The tool displays the 3D points projected on the camera image as an overlay in a different color channel so the operator can visually verify the calibration.

Evaluation of the coarse calibration was performed using the miscalibration error E_p which is computed both for the manual and the coarse calibration estimated using the rotated marker.

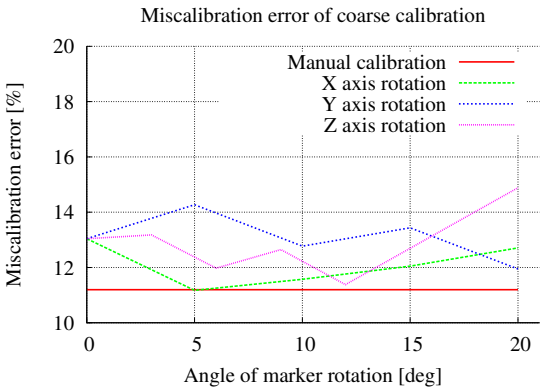


Figure 9: Miscalibration error (ratio of incorrectly projected points) of the coarse calibration against the manual one dependent on the rotation of the marker around the X, Y and Z axis.

Figure 9 presents the miscalibration errors E_P of parameters obtained through the process of the coarse calibration compared to the error of manually annotated parameters. The results prove that the camera-LiDAR pose estimation using the 3D marker yields similar miscalibration error as the manual calibration.

In comparison to the previously used calibrations techniques using chessboard markers [KP10, PMSE10, GMCS12], our approach with the 3D marker brings practical improvement – only a single frame from the camera and the LiDAR with a single marker appearance is needed for the sensors calibration. In contrary, the chessboard marker techniques require observations of the marker from various view points or a single observation of multiple marker instances placed in the scene.

Fine Calibration Results

After the initial coarse estimation of the calibration parameters, the refinement process follows in order to increase the calibration accuracy. The goal of this experiment is to evaluate impact of the refinement process on the precision of the calibration.

Figure 10 shows the different solutions evaluated by the edge cost function S_E (Equation (10)) proposed by Levinson and Thrun [LT13] and also by our novel miscalibration error E_P (Equation (11)) which may be considered as an objective factor evaluating each approach.

The solutions compared in Figure 10 use different cost functions for the calibration parameters evaluation dur-

 ISBN 978-80-86943-71-8

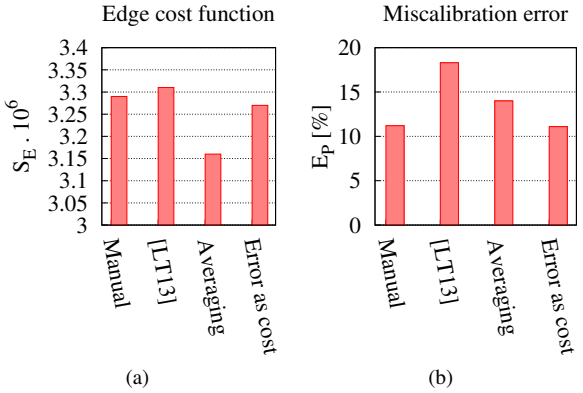


Figure 10: Refinement of the calibration results for the different evaluation criteria compared with the *manual* calibration. [LT13] maximizes the edge cost function S_E, the *Averaging* method computes the fine calibration as an average of all calibration vectors with higher S_E than the initial one. Miscalibration error E_P minimization is directly used in the search process of the last approach.

ing the search. This table also shows that the edge cost function S_E proposed by Levinson and Thrun [LT13] and used for refinement process does not correlate with the miscalibration error E_P. We would expect that the higher cost the solution has the lower error will be reached.

According to Figure 10, the best results were obtained using our projection error E_P criteria as the cost function for the calibration parameters. The drawback of using this error as a cost function is the requirement of the scene which has to be segmented into foreground and the background in both the LiDAR scan and the camera image.

The modification of the search process using the averaging of the all found calibration vectors better than the initial coarse one (see *Averaging* method on Figure 10) seems to be reasonable compromise between the calibration accuracy and the no requirement for the scene segmentation.

Figure 11 presents that the calibration of lower projection error (see Figure 10) yields also better visual results in the task of the point-cloud coloring. This fact also votes in favour of the miscalibration error rather then the edge cost function in objectivity of the calibration evaluation.

5 CONCLUSION

This paper presents a pipeline for the RGB camera calibration with Velodyne LiDAR. The first step in the calibration process estimates a coarse calibration using our novel 3D marker which allows to estimate the calibration precisely using a single pair of image-point cloud data and even a discplacement between the sensors is large (tens of centimeters).

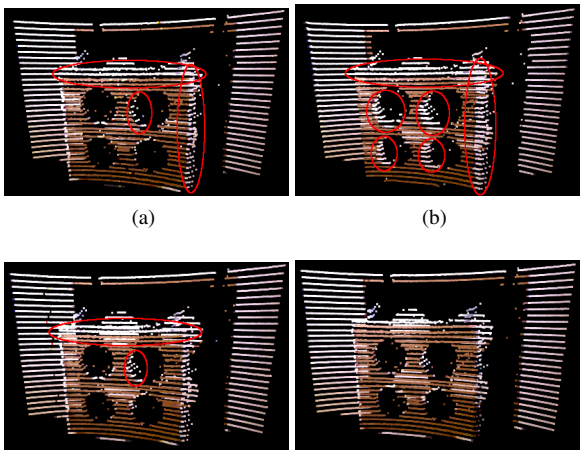


Figure 11: Coloured point cloud of our 3D marker when using the coarse calibration (a), results of refinement with edge cost function (b) (from [LT13]), our modification that averages all the parameters better that the initial one (c), and refinement computed using our projection error criteria as the cost function (d). The red ellipses highlight the largest errors caused by the inaccuracies of the calibration.

The consequent step refines the coarse calibration using a dense search in a small 6DoF calibration parameter subspace. The proposed modification of the calibration refinement process decreases the projection error by 5%.

The paper also introduces a novel miscalibration error metric for the calibration evaluation. This criterion comes from the basic calibration requirement that the points of particular object should be projected on the image segment occupied by this object. The objectivity of this evaluation function was also proved by the correlation with the visual quality of the point cloud colouring.

6 ACKNOWLEDGMENTS

The research leading to these results was funded by the TACR project V3C (no. TE01020415) and the IT4Innovations Centre of Excellence, grant n. CZ.1.05/1.1.00/02.0070, supported by Operational Programme Research and Development for Innovations funded by Structural Funds of the European Union and the state budget of the Czech Republic.

7 REFERENCES

[BCES11] Lionel Baboud, Martin Cadik, Elmar Eisemann, and Hans-Peter Seidel. Automatic photo-to-terrain alignment for the annotation of mountain pictures. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[BCK13] Yunsu Bok, Dong-Geol Choi, and In So Kweon. Generalized laser three-point algorithm for motion estimation of camera-laser fusion system. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2880–2887, May 2013.

[BJCK11] Yunsu Bok, Yekeun Jeong, Dong-Geol Choi, and InSo Kweon. Capturing village-level heritages with a hand-held camera-laser fusion sensor. *International Journal of Computer Vision*, 94(1):36–53, 2011.

[DBR09] B. Douillard, A. Brooks, and F. Ramos. A 3d laser and vision based classifier. In *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2009 5th International Conference on*, pages 295–300, Dec 2009.

[FB81] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, June 1981.

[GMCS12] A. Geiger, F. Moosmann, O. Car, and B. Schuster. Automatic camera and range sensor calibration using a single shot. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3936–3943, May 2012.

[HLT13] D. Held, J. Levinson, and S. Thrun. Precision tracking with sparse 3d and dense color 2d data. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 1138–1145, May 2013.

[KE95] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942–1948 vol.4, Nov 1995.

[KP10] A. Kassir and T. Peynot. Reliable automatic camera-laser calibration. In *Proceedings of the 2010 Australasian Conference on Robotics & Automation, ACRA 2010*, Dec 2010.

[LN09] Quoc V. Le and Andrew Y. Ng. Joint calibration of multiple sensors. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IROS'09, pages 3651–3658, Piscataway, NJ, USA, 2009. IEEE Press.

[LT13] Jesse Levinson and Sebastian Thrun. Automatic online calibration of cameras and lasers. *Robotics: Science and Systems*, XI, 2013.

[NMV10] J.I. Nieto, S.T. Monteiro, and D. Viejo. 3d geological modelling using laser and hyperspectral data. In *Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International*, pages 4568–4571, July 2010.

[Ots79] Nobuyuki Otsu. A Threshold Selection Method from Gray-level Histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, 1979.

[PMSE10] G. Pandey, J. McBride, S. Savarese, and R Eustice. Extrinsic calibration of a 3d laser scanner and an omnidirectional camera. In *7th IFAC Symposium on Intelligent Autonomous Vehicles (2010)*, pages 336–341, 2010.

[PMSE12] Gaurav Pandey, James R. McBride, Silvio Savarese, and Ryan M. Eustice. Automatic targetless extrinsic calibration of a 3d lidar and camera by maximizing mutual information. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 2053–2059, Toronto, July 2012.

[PPRJ13] Nunez P., Drews P., Rocha R., and Dias J. Data fusion calibration for a 3d laser range finder and a camera using inertial data. In *Proc. of 4th European Conf on Mobile Robots*, pages 31–36, 2013.

[ST94] J. Shi and C. Tomasi. Good features to track. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on*, pages 593–600, Jun 1994.

[TN13] Zachary Taylor and Juan Nieto. Automatic calibration of lidar and camera images using normalized mutual information. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, May 2013.

[YPIK90] H. K. Yuen, J. Princen, J. Illingworth, and J. Kittler. Comparative study of hough transform methods for circle finding. *Image Vision Comput.*, 8(1):71–77, February 1990.

[ZP04] Q. Zhang and R. Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). In *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2301–2306 vol.3, Sept 2004.