Improving Feature Matching Using Low Depth Resolution Range and Intensity Images

Viktor Kovács, Gábor Tevesz

Department of Automation and Applied Informatics,
Budapet University of Technology and Economics
Budapest, Magyar Tudósok krt. 2., Hungary
{Kovacs.Viktor, Gabor.Tevesz}@aut.bme.hu [1]

Abstract. In case of a mobile robot has to perform different tasks in an unknown environment it has to build its own map and at the same time localize itself in the map. This problem is known as SLAM (Simultaneous Localization and Mapping). In order to build a map or to perform localization landmarks must be detected. Various types of information may be used to extract landmarks or features. During our tests we used a structured light based range image sensor that also provides color images. The low depth resolution range images are smoothed and then planes are extracted. Viewpoint normalized patches are created and (SURF) features are extracted. These features serve as landmarks. We evaluate the benefit of viewpoint normalization.

Keywords: Range image, plane fitting, feature points, SLAM, viewpoint invariance.

1 Introduction

Considering a mobile robot moving in a known environment using a map, it has to be equipped with the knowledge of data association in that specific context to recognize and match sensed information to its map. Based on the priori and sensed information it is able to localize itself. In case the robot is moving in a previously unknown environment it has to create the map while also localize itself maintaining a consistent map. SLAM (Simultaneous Localization and Mapping) theory deals with this problem.

The basic idea behind SLAM is to do measurements of landmarks from different viewpoints, gradually decreasing the uncertainty of the position of the landmarks relative to each other and updating the position estimate in each step. The uncertainty of the sensor and the motion model has to be known. There are two major implementations: the EKF (Extended Kalman-Filter) based solution and the particle filter based.

[1] A major part of the paper has already been reviewed and published in Proc. of Automation and Applied Computer Science Workshop 2012. pp. 169.

In this paper we deal with camera image based landmark detection. To find landmarks we utilize image feature extraction algorithms. These algorithms first return points that are considered extremal points in a sense and invariant to some transformations such as translation, rotation, scale and illumination. In the second pass descriptor vectors are formed from the image content around the previously found keypoint candidates. The image content is considered similar in different images or different parts of images if the distance (Euclidean) between descriptor vectors is small. Descriptors are ideally invariant to transformations such as translation, rotation, scale, brightness, contrast, viewpoint etc. Advanced image feature extraction algorithms such as SIFT [5] and SURF [9] tolerate these transformations very well.

As a mobile robot moves in its environment, it is essential to achieve good viewpoint invariance to match landmarks observed from large viewpoint changes. Considering stable landmarks that we are able to match robustly from different viewpoints, a lower number of landmarks are needed to operate SLAM which is a benefit considering the size of the covariance matrices in the EKF algorithm.

In this paper we show the benefits of using range and intensity images combined for landmark detection. We use the range image to find plane surfaces where the intensity image shall be normalized. We use a feature detection algorithm to extract and match features from normalized images. We present results from both simulated and sensor based images.

In Section 2 we present related work, briefly overview some feature detection algorithms and provide references for more detailed descriptions. In Section 3 we show the proposed algorithm and the framework used to create simulation data and evaluate results. In Section 4 we present the results based on both simulation and real world sensor data. Finally we draw conclusion and give plans for further research.

2 Related work

In this section, related work is discussed considering the building blocks of the examined algorithm such as image feature detectors, improvements to feature detectors and their extensions, hole filling algorithms, viewpoint normalization.

2.1. Feature detectors

Features are unique patches in images which can be redetected even if the image is transformed thus invariance is important. One of the basic feature detectors is Harris corner detector [3], [4] . Edges describe image content very well and are less sensitive to illumination changes but suffer from the aperture problem. That is why corners where edges meet or break is considered much better candidates. Harris corners are only keypoints, the algorithm does not propose a specific descriptor. Harris corners are relative dense in a picture, while being rotation, translation invariant and rejects some changes in illumination and scale. These feature points may be used to find correspondences between consecutive image frames. A custom descriptor may be used to enable matching.

Scale Invariant Feature Transforms (SIFT) was introduced in [1]. Both the keypoint detection and descriptor formation were proposed. Speeded Up Robust Features (SURF) [2] provides similar results but there are major differences how the algorithms are implemented [5].

The algorithms mentioned above do not utilize all the information in the image. Harris corners, SIFT, SURF only use luminance of the image. There are many additions to these algorithms to improve distinctness of the descriptors. In [6] a method is proposed to take advantage of color information for descriptors making feature point detection and matching more stable. In [7] a comparison may be found of several color utilizing feature extraction algorithms. It is shown in [8] how color invariance was added to SURF. In [9] an enhancement is described how to enhance the SIFT algorithm to be affine invariant, although this comes with a huge price in computation expense as it evaluates the feature points at multiple virtual camera views.

2.2. Hole filling

In the proposed algorithm we utilize the algorithms used in mesh hole filling to improve the low depth resolution range images for further processing. In [10] a method is described for filling holes in surface meshes by locally fitting quadratic parametric surfaces. Parameter estimation is based on weighted least squares algorithm, vicinity points around holes are used to estimate the linear parameters of the model.

2.3. Plane detection in range images

There are different approaches for range image segmentation. A comprehensive review can be found in [11]. Edges, boundaries of objects are easy to detect in range images and similar tools are needed just like for edge detection in intensity images. Edge detection in intensity images suffer from many defects such as false edges may be caused by texture and lighting. In range images different effects appear. Gradual surface normal direction change might not appear as discontinuity in depth values but this segmentation is not suitable for plane detection. In many cases surface curvature is calculated for each pixel, allowing segmentation for curved surfaces maintaining viewpoint invariance.

Edge detection is a good start for plane detection, but segmented surfaces must be evaluated whether these are planar surfaces or not. Also edges might not surround a region completely, which is a problem need to be dealt with. Region growing algorithms are also used to iteratively add pixels to regions considered planes. Other methods such as PCA-RANSAC may be used to evaluate whether a set of points are considered as a plane or what set of points fit best to a plane. While RANSAC is designed to work well in large percentage of outliers, it is worth utilizing local information stored in range images.

Another method for plane detection is to create a histogram of local surface normal vectors, as the normal vector of a planar surface point in the same direction, peaks

appear in the histogram at the specific areas associated to the characteristic normal directions. Further segmentation is needed to separate different planar regions having the same normals. As local surface normal approximation is sensitive to noise such as edge detection both being based on difference operator, a proper preprocessing is needed of the images. Low depth resolution images contain large quantization error which must be dealt with.

Further methods such as Hough transform can also be used for plane detection, but real-time application is limited due to high computation and memory capacity requirements. Boundary conditions may be introduced to improve its efficiency.

2.4. Viewpoint normalization

The idea is not novel, there have already been papers published in the field. In [12] Viewpoint Invariant Patches were introduced which are local descriptors for textured surfaces formed by projecting the texture on a fitted local plane and SIFT descriptors are generated in this image. Dominant planes in urban environments were detected using vanishing points [13] and laser scanners [14], and used these normalized images to improve SIFT viewpoint invariance.

3 Implementation of the proposed algorithm

A framework was developed for the evaluation of algorithms. It is able to generate images by simulation and connect to the range sensor and process the images. Simultaneously captured intensity and range images are shown in Figure 1. The two images from the sensor do not overlap correctly. A 2D transformation (translation, scale, rotation) was estimated to match the images.

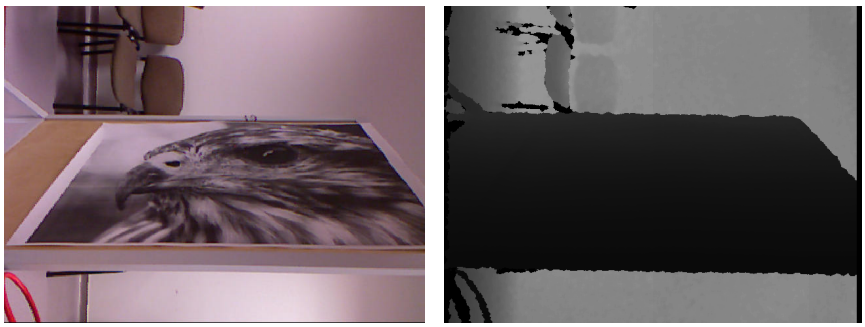

Fig. 1. (left) Captured intensity image (right) corresponding range image from the Kinect sensor.

3.1. Simulation

A framework had been developed to generate and process simulation data. First simulation data was generated as ideal range images, not dealing with quantization error. These ideal range image were processed in the following way: for each pixel a local surface normal vector was estimated based on neighboring pixels. Two components of the 3D normal vector for each pixel was collected in a histogram as the third component is redundant due to the vectors had been normalized.

Characteristic planes appeared as peaks in the histogram space allowing easy identification and segmentation. For each characteristic plane a viewpoint normalized image was generated. Image features were extracted from these normalized images. For comparison purposes image features were also extracted from the original images. A batch evaluation was carried out by changing the view angle gradually and the number of feature matches were evaluated.

3.2. Captured sensor data

Captured real-world data suffer from numerous errors such as noise and quantization error. Quantization error makes it difficult to detect planes in range images. As continuity is broken, large planes appear as lane strips behind each other perpendicular to the camera. In this section we describe how preprocessing is done to smooth range images, detect planes and normalize planar intensity patches.

3.3. Range image smoothing

Range or depth images encode distance information for every pixel. Based on the sensor used (time of flight range finders or structured light based) it may encode the distance from the sensor or the Z coordinate (depth) of a measured point.

First low depth resolution images must be smoothed to apply further processing. The range image is decomposed into planar patches with uniform depth values using the connected component labeling algorithm. For every edge pixel of these patches so called vicinity points are calculated by the mean world coordinate positions with adjacent pixels of neighboring patches. In case adjacent neighbors have larger difference in depth dimension than the estimated quantization error, those patches and its neighbors are omitted. To estimate an updated pixel value local second order surface is fitted by parameters estimation based on surrounding vicinity points. By defining a recursion number a set of neighboring patches are selected (empirically determined 1-3 levels).

Vicinity points of the surrounding points are used only in a defined distance threshold. This distance is a function of the observed depth value. As the density of measurement points decrease by the distance the feature helps to overcome this shortcoming. The restriction is needed as patches may be very large across the image and surface should be estimated from local data. In case the number of the selected controlpoints is still large, a subset is randomly chosen and surface parameter estimation is done based on these values using the linear LS method.

The Z depth values of pixels contain measurement noise and even more importantly quantization noise. Unfortunately calculated X and Y coordinates depend on the measured depth value, so estimating new depth values for pixels modify the original values used for surface fitting. These vicinity points are considered to be good approximation of quantization error eliminated values.

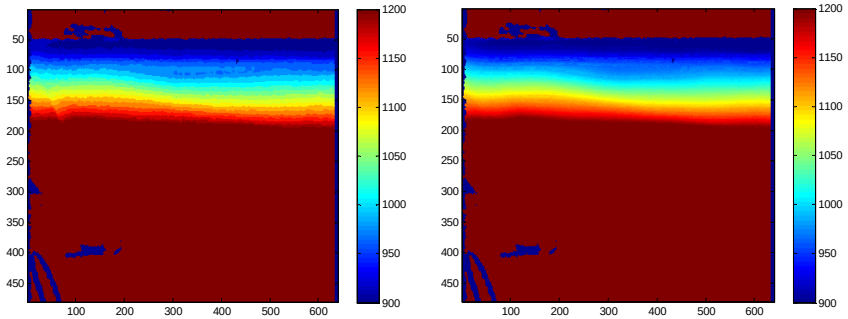

Fig. 2. (left) Original range image, (right) filtered range image.

Perspective-corrected Sobel convolution filters of size 5x5 are used to determine the local surface tangent vectors for each pixel. From these vectors, the normal vector is formed by cross product. Figure 3 shows the x and y components of the approximated normal vector. As one component of the normal vector is redundant and n_x, n_y components are not linear in the angle of the surface, for each pixel an ($\alpha_x=sin^{-1}(n_x)$; $\alpha_y=sin^{-1}(n_y)$ pair is calculated. A histogram is formed of these α_x and α_y values.

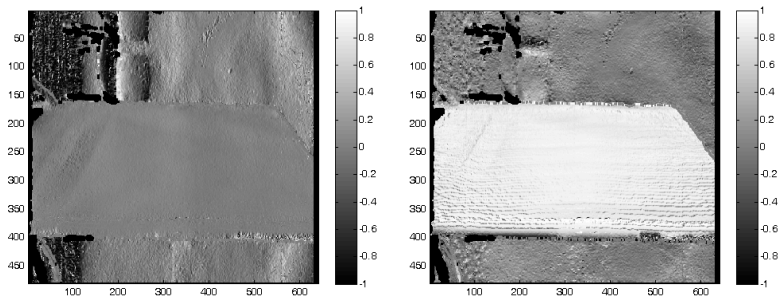

Fig. 3. (left) Local surface normal x component, (right) y component.

A simple region growing segmenting algorithm is used to separate peaks in the histogram. Bounds are fixed (-π..π), and arbitrary bin number is used (181x181). Figure 4 shows the histogram and the regions associated to separate peaks. Detected planes are presented in Figure 5 based on the histogram regions. Scattered outliers are removed by applying a mode filter.

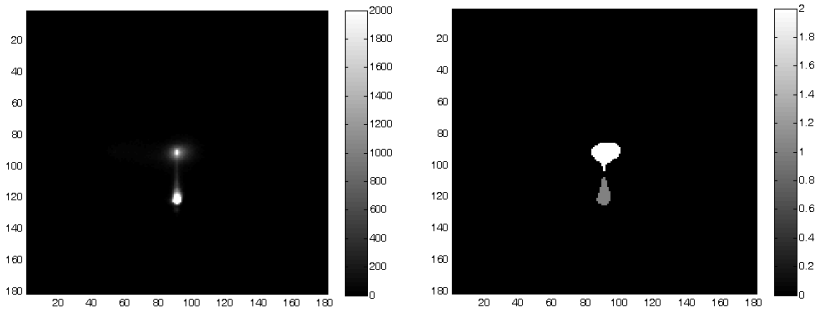

Fig. 4. (left) Peaks refer to regions facing the same direction, (right) segmented peaks detected in the histogram.

$$p = \arg\min_{p_i}((\alpha_x(p_i) - \overline{\alpha_x})^2 + (\alpha_y(p_i) - \overline{\alpha_y})^2) \quad (1)$$

In case the image contains multiple parallel planar patches having normal vectors pointing the same direction these patches would be put into one segment bin. To overcome this shortcoming the following is being done. Average $\overline{\alpha}_x$ and $\overline{\alpha}_y$ values are calculated for the largest histogram bin in the segment. A *p* point is selected which minimizes Eq. (1).

A transformation T_1 is formed from translating by -*p* and rotating by $\overline{\alpha}_x$ around the *y* axis and by $\overline{\alpha}_y$ around *x* axis. T_1 transformation is applied to all points of the current histogram segment. As a result the plane(s) are transformed parallel to the *xy* plane. In case there were multiple patches with the same orientation these are parallel to *xy* also. Next a histogram is generated by projecting the point cloud on the *z* axis. Different parallel planes having different *z* values appear as peaks in the *z*-histogram. A simple segmentation algorithm is used to segment parallel facing planes. The boundaries of the histogram is [-5000..0mm] having 50 bins.

The algorithm first looks for the bin having the highest value and creates the first *z*-segment. Neighboring bins are added until bins are found with values less than a threshold. In this case the algorithm starts again and forms the second segment starting with the highest bin having larger value than the threshold. This algorithm is applied to all previously found α_x; α_y histogram segments. New segments are rejected if the sum of pixel number is lower than a threshold, meaning the planar patch is not large enough.

For every updated segment a viewpoint normalized image is generated. Segmenting by the *z* direction results in creating multiple normalized images even if the planes are parallel. In case only one image would be generated the farther planes would appear smaller having lower resolution making feature extraction algorithms not as efficient. This is why multiple normalized images are generated, allowing to utilize the arbitrary resolution (640x480) the best at which the viewpoint normalized images are rendered.

For each z-segment a transformation T_2 is evaluated. After having the pixels transformed by T_1 the boundaries (*min(x), min(y), max(x), max(y), max(z)*) are evaluated. T_2 consists of a translation by (*-min(x), -min(y), -max(z)*) and a scaling by (*w/(max(x)-min(x)), h/(max(y)-min(y)), 1*) to normalize the points to image coordinates.

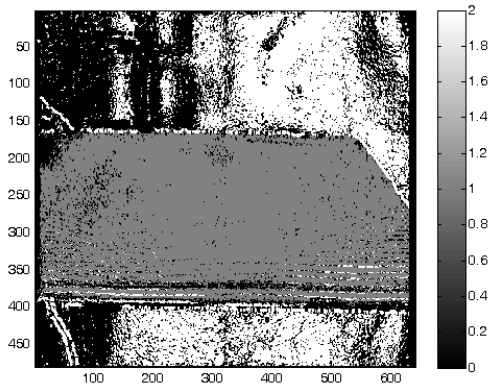

Fig. 5. Final segmentation of the image to planes.

Finally to generate the normalized image, *u* and *v* are iterated through the pixels coordinates of the image to be rendered. (2) is used to transform pixel coordinates to world coordinates. These coordinates are then projected to the source image coordinate system (3) to acquire the original pixel from the source image that is processed where *W* is the destination image width in pixels, *H* is the height, *fov* is the field of view. Bilinear interpolation is used to map source to destination pixels.

$$w = (T_2 T_1)^{-1}(u, v, 0)^T \tag{2}$$

$$s_x = \left(\frac{w_x}{tan(\ fov/2)z} + 1 \right) \frac{W}{2}; s_y = \left(\frac{w_y}{tan(\ fov/2)z} \frac{W}{H} + 1 \right) \frac{H}{2} \tag{3}$$

Fig. 6. (left) Normalized image to the background wall (right) normalized to the rotated plane.

As a result of the algorithm using the previously shown test images (Fig. 2.) two viewpoint normalized pictures were generated (Fig. 6.). In the example one corresponds to the wall in the background, the other is generated based on the rotated plane.

3.4. Feature extraction and matching

For feature extraction the SURF algorithm had been chosen. There are several implementations, first the OpenSURF was utilized. OpenCV, the commonly used open computer vision library, also has implementations for SIFT, SURF and other feature point extraction algorithms. A comparison of how both algorithms benefit from the viewpoint normalization is a future work issue.

The OpenSURF localizes keypoints and generates descriptor values based on the surrounding data around the keypoints. Matching is evaluated by determining the Euclidean distance between descriptor vectors. In case the distance is lower than a threshold, it is considered a match. During simulation it is easy to evaluate whether a match is true or false positive. In real world tests a transformation between the normalized image keypoints and the reference image keypoints may be determined. In case of a high number of false matches RANSAC method is commonly used to reject outliers. This transformation can be used to determine whether matches are correct or not.

4 Results

In this section we present results obtained from simulation and from real world captured data. Latter results are based on a different plane detection (LS-RANSAC in point cloud) and normalized method although results are suspected to be similar. Feature extraction and matching with the previously shown algorithm is a subject of future work.

4.1. Simulation

Simulation data do not suffer from quantization noise so it is quite straightforward to estimate surface normals and generate the normal orientation histogram and segment the bins.

Two test were conducted. The first contained only one plane which was rotated in 3D by two angles around x and y axes.The second test was based on two instances of a simple scene consisting of three planes that were rotated in opposite directions and the planes were extracted, viewpoint normalized and matched between the scenes.

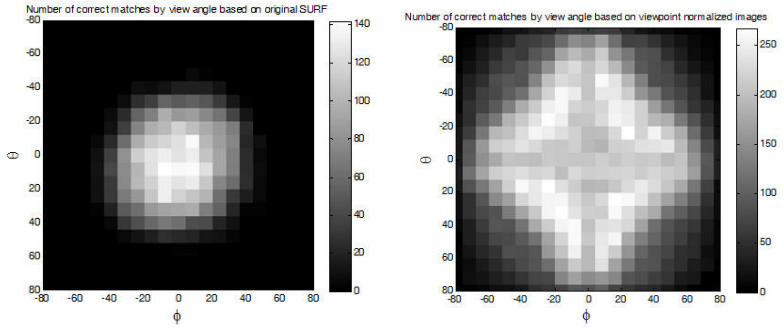

Fig. 7. (left) Without viewpoint normalization (right) with viewpoint normalization.

Results of the first test is shown in Figure 7. It presents the number of correct matches as a function of view angle based on the original method without and with viewpoint normalization. It is visible that there is a benefit providing larger view angle difference while improving feature matching. The findings are consistent with the literature where it is shown that SURF provides valuable matches until the viewpoint changes approximately 30-35°, the normalization has large benefit expanding these angles to 50-70°.

4.2. Captured sensor data

Results shown here are generated by LS-RANSAC plane fitting method. As the algorithm is based on random sampling, it was not as reliable in finding the appropriate plane as the algorithm presented in this papery. Although the plane detection differs, the results (viewpoint normalized images) are expected to be similar thus giving similar results. The combination of the previously shown algorithm and SIFT, SURF feature detector comparison is a subject of future work. A textured plane was rotated gradually by 15 degrees. The number of matches by simple SURF and number of matches using normalization is shown in Table 1.

Table 1. Results based on sensor data.

Angle	-75°	-60°	-45°	-30°	-15°	15°	30°	45°	60°	75°
Original	0	6	57	124	156	158	112	26	0	0
Normalized	31	106	98	146	160	149	148	107	57	5

5 Conclusion and Future work

Range image smoothing, normal vector orientation histogram based segmentation and image feature matching based on viewpoint normalization were shown in this paper. It was presented that additional information gained of the local geometry improved

matching rate and stability. Future work includes feeding the normalized images based on the algorithm mentioned in this paper to the feature matching algorithm. Also optimization and further research will be carried out to investigate the benefit of the method to more feature extraction algorithms.

Acknowledgments

This work was partially supported by the European Union and the European Social Fund through project FuturICT.hu (grant no.: TÁMOP-4.2.2.C-11/1/KONV-2012-0013).

References

1. D. G. Lowe, “Distinctive image features from scale-invariant keypoints”, Int. J. Comput. Vision, vol. 60, pp. 91-110, Nov. 2004.
2. H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-up robust features (surf),” Comput. Vis. Image Underst., vol. 110, pp. 346-359, June 2008.
3. C. Harris and M. Stephens, “A combined corner and edge detector,” in Proceedings of the 4th Alvey Vision Conference, pp. 147-151, 1988.
4. K. Mikolajczyk and C. Schmid, “Scale & affine invariant interest point detectors,” Int. J. Comput. Vision, vol. 60, pp. 63-86, Oct. 2004.
5. J. Bauer, N. Sunderhauf, and P. Protzel, “Comparing several implementations of two recently published feature detectors,” in Proceedings of the International Conference on Intelligent and Autonomous Systems, IAV, Toulouse, France, 2007.
6. A. E. Abdel-Hakim and A. A. Farag, “Csift: A sift descriptor with color invariant characteristics,” in CVPR (2), pp. 1978-1983, IEEE Computer Society, 2006.
7. K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, “Evaluating color descriptors for object and scene recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 9, pp. 1582-1596, 2010.
8. D. M. Chu and A. W. M. Smeulders, “Color invariant surf in discriminative object tracking,” in ECCV Workshop on Color and Reflectance in Imaging and Computer Vision, 2010.
9. G. Yu and J.-M. Morel, ”ASIFT: An Algorithm for Fully Affine Invariant Comparison,” Image Processing On Line, 2011.
10. J. Wang and M. M. Oliveira, “A hole-filling strategy for reconstruction of smooth surfaces in range images.,” in SIBGRAPI, pp. 11-18, IEEE Computer Society, 2003.
11. S. Bose, K. Biswas, and S. Gupta, “An integrated approach for range image segmentation and representation,” vol. 10, pp. 243-252, August 1996.
12. C. Wu, B. Clipp, X. Li, J.-M. Frahm, and M. Pollefeys, “3D model matching with viewpoint invariant patches (VIPs),” CVPR08, 2008.
13. Y. Cao and J. McDonald, “Viewpoint invariant features from single images using 3D geometry.,” in WACV, pp. 1-6, IEEE Computer Society, 2009.
14. Y. Cao, M. Yang, and J. McDonald, “Robust alignment of wide baseline terrestrial laser scans via 3D viewpoint normalization,” pp. 455-462, 2011.