Improving viewpoint invariance of image feature extraction methods using intensity and range images

V. Kovács, G. Tevesz
Department of Automation and Applied Informatics
Budapest University of Technology
Budapest, Hungary
kovacsv@aut.bme.hu, tevesz@aut.bme.hu

Abstract—**The most common image feature extraction algorithms such as SIFT (Scale Invariant Feature Transform) and SURF (Speeded Up Robust Feature) have been proven to be invariant to changes in rotation, scale and with restrictions to illumination and viewpoint changes. These algorithms generate descriptor vectors around keypoints in 2D images. Close descriptors suggest similar image patch. In case of mobile robotics applications it is important to achieve good viewpoint invariance and stability to detect landmarks and objects with high reliability. Improving viewpoint invariance for image feature detection increases the efficiency of SLAM algorithms. In this paper we present and evaluate a method to use additional data provided by range image sensors to supplement traditional feature extraction algorithms to improve viewpoint invariance. We present the method and results of computer simulation and also real world examples comparing the SURF (OpenSURF) with and without the improvement. An active structured light based range and intensity image sensor was used to acquire real world test images.**

Keywords-component; image feature extraction; landmark detection; object recognition; viewpoint invariance

I. INTRODUCTION

Machine vision algorithms especially in autonomous mobile robotics applications require highly reliable feature detection and extraction algorithms. When an autonomous mobile robot moves in a known environment, visual localization can be based on the detection of objects and landmarks. When a robot is placed in an unknown environment it has to build a map and localize itself in the map at the same time. The theory concerned with this problem is SLAM (Simultaneous Localization and Mapping). The basics were elaborated over twenty years ago but the data association problem such as detecting and measuring landmarks may still be improved.

Image feature extraction methods are great tools for identifying objects in an image or match different image patches. The most common feature extractors such as SIFT [3] (Scale Invariant Feature Transform) and SURF [7] (Speeded Up Robust Feature) provide robust feature detection independently of changes in translation, rotation and also reject small changes in illumination and viewpoint variance. These algorithms use a 2D intensity image as input and return a collection of feature points and descriptors. Descriptors are vectors formed from the image around the feature points. Short Euclidean distance between feature descriptors suggests similar image information. This can be used for finding correspondences between keypoints in images or object detection (using a database of descriptors of object images). In case of mobile robotics applications it is important to achieve good viewpoint invariance and stability to detect landmarks and objects with high reliability. SLAM theory requires to detect and measure the position of landmarks. Improving viewpoint invariance for image feature detection increases the efficiency of SLAM algorithms.

In this paper we evaluate a method to improve viewpoint invariance of feature detectors by utilizing additional information provided by range sensors. As mobile robots move around, viewpoints vary all the time. It is advantageous to improve viewpoint invariance to recognize the same landmarks or objects from even wider angles. First we present the method and the software framework used to evaluate the original and the improved method by simulation. Later we show real world examples to give numerical results of comparative tests.

II. RELATED WORK

A. Feature Detection Algorithms

There are numerous types of feature detectors each have their advantages and disadvantages. It is important to select the one best suited for a specific task. Specification may include the number of feature points detected, the ratio of correct matches, type of invariance, complexity of the algorithm. A detailed comparison may be found in [10].

Scale Invariant Feature Transforms (SIFT) was introduced by Lowe [3]. First the image is transformed to scale space by applying different convolution with Gaussian kernels using different σ scale parameters (1). Difference image (D) is calculated between neighboring filtered images (3).

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y) \tag{1}$$

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{2}$$

$$D(x,y,\sigma) = L(x,y,k\sigma) - L(x,y,\sigma) \tag{3}$$

Where I is the original image, G is the Gaussian filter for generating scale space of the image and L is the convolved image in scale space.

Local extrema is searched in neighboring pixels in spatial and scale dimensions (to achieve scale invariance) in image D. These extremas are considered as keypoint candidates. Low contrast and edge keypoints are rejected as being insufficiently stable. An orientation histogram (with 36 bins) is formed from the neighboring pixels in the closest scale image. Dominant direction is considered as the orientation of the keypoint (rotation invariance). Local descriptors are assigned to keypoints to enable feature point matching.

Descriptors are formed by creating histograms (for example 8 bins) around the keypoints from weighted gradients. For example 4x4 histogram arrays are calculated around the keypoint, each containing 4x4 gradient values normalized to the feature orientation. A descriptor vector is formed from the histograms (4x4x8 values). This vector is normalized to achieve contrast invariance, gradients are unaffected by brightness changes. Nonlinear illumination effects are rejected by reducing the influence of peaks by thresholding the values in the vector to a value (0.2). Finally the descriptor is normalized again.

The SURF (Speeded Up Robust Features) uses a different approach. The partial second order derivative of the Gaussian in discrete space is approximated by box filters. Instead of reducing the image size the box filter kernel is increased. To improve performance, the image is first transformed to integral space where calculation speed is independent of the filter size. Localization of interest points are done by finding the maxima of Hessian matrices. Haar wavelet filter responses are calculated in both x and y directions around the keypoints. These responses are weighted by a Gaussian and the most significant direction, which is calculated along a rotating window is considered as the orientation of the keypoint.

Square regions that's size is dependent on the scale are constructed around the interest points and are rotated by the orientation of the feature point. These squares consist of subregions to preserve spatial information such as SIFT does. Haar-wavelets are constructed and evaluated in both x and y directions relatively to the orientation in a defined number of points. From these responses inside a subregion $\{\Sigma dx, \Sigma|dx|, \Sigma dy, \Sigma|dy|\}$ vector is formed. The descriptor vector is created by concatenating these vectors of all subregions and normalizing it.

Both algorithms serve the same purpose but the implementation is approached from a different viewpoint. Each one has its advantage.

In [8] different implementations of feature detectors including the SIFT and SURF algorithms are compared. Rejection of different transformations are evaluated such as rotation, change of scale, added noise, change of lighting conditions and viewpoint. According to their findings SIFT performed best in the total number and ratio of correct matches however SURF followed very closely. An advantage of the SURF algorithm is the lower computational cost. This is a benefit in real-time applications. According to the evaluation of the effect of viewpoint change, the number of correct matches drops significantly above 10°. Above 30-35° no correct matches were identified.

B. Improvements of feature detection algorithms

There are different improvements to the base feature detector algorithms.

The original SIFT implementation utilizes only the luminance information stored in images. Color information is neglected. [1] proposes a method to take advantage of the color information for descriptors allowing the algorithm to enhance feature point detection and matching stability. [9] evaluates a number of feature detectors that makes use of color information. SIFT can be used for feature detection independently for each channel in different color spaces. [9] recommends using OpponentSIFT based on the opponent color space. Similar extensions of the SURF algorithms were also published. A comparison between different methods for utilizing color information by the SURF algorithm may be found in [4].

PCA-SIFT [14] was introduced to reduce the size of the descriptor vector by principal component analysis while still maintaining the ability to distinguish descriptors.

Affine invariant feature detectors were also developed [11] [12]. The ASIFT extension [6] suggests normalizing images by simulating different viewpoints of an affine camera model. This method provides very good results however there is no information about the camera pose thus experimenting with different parameters consumes much processing capacity.

C. Related work

Viewpoint-Invariant Patches (VIP) were introduced in [2]. VIPs are extended local descriptors for images with known 3D geometry. The construction of VIPs start by constructing local orto-textures by projecting pixels to a local tangent plane. SIFT keypoints are detected in this texture. The SIFT descriptor is complemented by additional information such as: scale, surface normal, local gradient orientation. This information is sufficient to define the 3D transformation between matching VIPs.

In [13] and [16] laser scanner was used to acquire 3D point cloud and a hand-held camera to capture normal intensity data. Dominant planes are identified in the point cloud and image rectification is done to create viewpoint invariant patches. Plane detection is based on RANSAC and MDL (minimum description length). SIFT features are extracted from these patches. The complete viewpoint invariant feature contains the position in 3D space, position in the image, the patch scale, the dominant gradient orientation of the patch and the 128 element SIFT descriptor. The method was used to match captured data in urban environments. In [15] a similar method was shown but instead of capturing 3D data, local geometry was obtained by line segments and vanishing points. Considering these restrictions this is feasible in urban environment as artificial structures contain straight and parallel lines such as edges and material boundaries. These lines define vanishing points which are used to identify homography transformation.

III. IMPLEMENTATION OF THE FRAMEWORK

A framework was implemented to evaluate the performance of the algorithm. Software supports both simulation and processing of real world captured data.

A. Simulation

To create simulation inputs a 3D viewport was used to render a 2D image in 3D space from different viewpoints. Also the same framework was used to generate the range image by utilizing the 3D hit test features. Intensity images were rendered with optional supersampling to avoid aliasing effects.

$$\begin{pmatrix} u' \\ v' \\ w' \\ 1 \end{pmatrix} = \left(\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ \hline 0 & 0 & 0 & 1 \end{array}\right) \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{4}$$

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u'/w' \\ v'/w' \end{pmatrix} \tag{5}$$

Perspective mapping consists of two steps. First applying a linear transformation to rotate and translate the points to the camera coordinate system (4). Then applying a projection according to the pin-hole camera model (5).

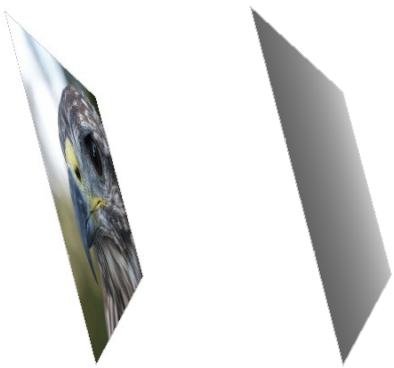

Figure 1. Simulated 3D intensity and range images.

$$\begin{aligned} x &= tan\left(\frac{\alpha}{2}\right)\left(\frac{2u}{W}-1\right)z \\ y &= tan\left(\frac{\alpha}{2}\right)\left(\frac{2v}{H}-1\right)\frac{H}{W}z \end{aligned} \tag{6}$$

Knowing the field of view (α) and the resolution (W, H) of the image and the Z coordinate for each pixel (u, v) from the range image a 3D point (x, y, z) cloud may be reconstructed (6). The normal intensity image provided colors for each point in the cloud. In simulation there was no need to calibrate the intensity and the range images as both were taken from the same viewpoint. After retrieving the point cloud a best fit plane was identified. We use the RANSAC (Random Sample Consensus) algorithm to identify best fit models. The algorithm selects m random samples from n input samples where $m \ll n$.

The fitter function is used to evaluate the model (7) parameters from the random samples. In the next step all the samples and the determined model parameters are used to evaluate the cost of the fit. This process is repeated for i iterations. The best overall model parameters are returned. For our tests we use least squares (8) method for fitting a plane as it requires much less computational effort than PCA and we consider to have measurement error in the Z direction (depth) not in X and Y as the image was sampled equidistant.

$$z = \vartheta\begin{pmatrix} x & y & 1 \end{pmatrix}^T = \vartheta_1 x + \vartheta_2 y + \vartheta_3 \tag{7}$$

$$\vartheta_{LS} = \left[X^T X\right]^{-1} X^T Y \tag{8}$$

where $X_i = \begin{pmatrix} x_i & y_i & 1 \end{pmatrix}$ $Y = \begin{pmatrix} z_1 & z_2 & \dots \end{pmatrix}$ pixel coordinates and corresponding depth values.

The detected plane is described by its normal vector and the its center of gravity. The orientation of the normal vector is given but the direction is not obvious. Depending on where the plane is identified in the image the direction of the normal vector must be changed according to eq. (9). The dot product of the normal vector and a vector pointing to a point in the identified plane must not be negative. The direction of the normal vector must be changed to satisfy the condition.

$$n \cdot p \geq 0 \tag{9}$$

After the direction is corrected we identify the transformation matrix that transforms the fitted points on the plane determined by the Z axis.

$$n_1 = (0,0,1) \tag{10}$$

$$a = n_1 \times n_2 \tag{11}$$

$$w = a \times n_2$$

$$\beta = \measuredangle\ n\ \ n\ \ = atan2(w \cdot n_1, n_2 \cdot n_1) \tag{12}$$

After identifying the bounding box of the fitted point cloud the inverse transformation is used to render the original distorted image from a viewpoint that is parallel to the normal vector of the fitted plane (10) (at arbitrary resolution). The image has to be rotated around a axis (11) by β (12). We consider this as the viewpoint normalized image. Sampling is done via nearest neighboring pixels. This image contains the same amount of visual information as the distorted but the additional information provided by the range image is used to modify the distorted image to create a new one closer to the original.

During our evaluation we used the SURF [7] algorithm to identify feature points in images. The OpenSURF[1] implementation was used which is available in C++ and C# sources. We ran the algorithm on the original undistorted and both the distorted and the normalized images. The normalized image is translated, rotated and scaled compared to the original image but as the feature detectors reject these transformations the chances are good to find matches with the original image.

[1] http://www.chrisevansdev.com/computer-vision-opensurf.html

For evaluation purposes we use high probability matches (close Euclidean distances such as 0.1-0.15 according to our experiences) to determine the rotation, translation and scale parameters. The inverse transformation is applied to render an image that has the same orientation, scale and position as the original one. Detected features in the reconstructed image are transformed on this image to visually help evaluation. This way the matches can be easily seen and false matches can be simply identified automatically.

B. Capturing real world images

To collect both intensity and range images we used the commercially available low cost sensor from Microsoft. It incorporates a traditional webcam and an IR active structured pattern based range sensor. Both sensors provide images of resolution 640x480, 30 frames per second. The field of view is 57° horizontally and 43° vertically.

The two images are taken from a slightly different view, and by different view angle. To compensate this we recorded point correspondences to determine the parameters of a simple model (13) to fit the two images.

$$\begin{pmatrix} x_m \\ y_m \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} c_x \\ c_y \end{pmatrix} \qquad (13)$$

The SDK[2] incorporated a proprietary method of calibration but this was not used considering the lack of inverse transformation. Only the raw images provided by the sensor were used. The method of image processing was the same as shown in simulation.

IV. RESULTS

Results based on simulation and captured image data is presented in this section.

A. Results based on simulation

A number of simulation were run in batch mode modifying viewpoint in each case. An 800x600 pixels size image was used as a reference. It contains both soft and high contrast transitions, repeating textures (feathers). This image was rendered as a plane image in 3D from different relative viewpoints. The viewpoint varied between -80° to +80° in 21 steps. Both generated images were rendered to resolution of 640x480. The intensity image was oversampled to eliminate artifacts caused by aliasing.

After rendering the SURF algorithm was applied and descriptors were searched. Figure 2. shows the number of correct matches that were found in the original image. As the view angle approaches the extrema the number of matches decrease gradually. Matches are determined by calculating the Euclidean distance between descriptors. A match is found if this value is < 0.15 (determined empirically). To differentiate between false and correct matches we used the inverse of the transformation that was used to render the images for simulation. The coordinates of the feature point found in the perspective distorted image is transformed back to the original image coordinate system. If the distance is smaller than a threshold (10 pixels) the match is correct. The algorithm provides valuable matches between approximately ±30°-35° changes. This is consistent with the findings in [7] and [8].

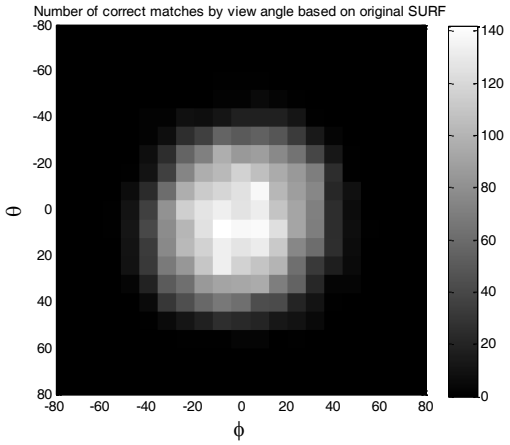

Figure 2. Number of correct feature matches by the original OpenSURF implementation as the function of viewpoint orientation (ϕ, θ : -80°..+80°)

Viewpoint normalization method first creates a point cloud from the range image. It uses the RANSAC-LS method to find the best fitting plane. It identifies the transformation to rotate the plane perpendicular to the Z axis. The bounding rectangle is identified and the normalized image is rendered by utilizing the inverse transformation. This image has an arbitrary resolution, during the tests 640x480 pixels were used. At extremal view angles loss of data is easily noticeable in Figure 3.

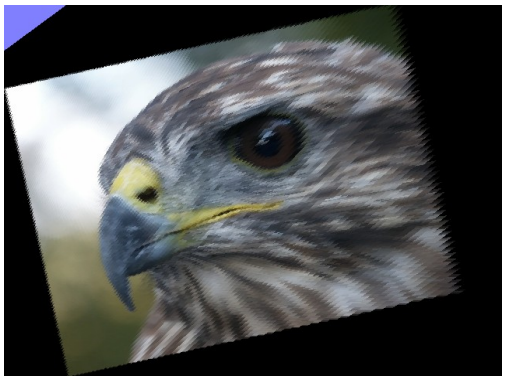

Figure 3. Degredation of quality in the normalized image.

SURF descriptors are searched in the normalized image. Matching is executed similarily as by the normal method mentioned earlier. To separate false and correct matches we identify the 2D transformation (translation, rotation and scale) that transforms the matches between the normalized image and the original. The matching descriptors are transformed and the distance is evaluated. Matching distances outside of a threshold (10 pixels) are considered false matches. Figure 4. shows the number of correct matches as the function of view angles around both axes. Results show that correct matches were found over wider view angles. There are still a number of matches found over 50°-60° view angles, showing much better viewpoint invariance than in the previous case. Actual values vary by image content.

[1] The SDK is available from http://kinectforwindows.org/

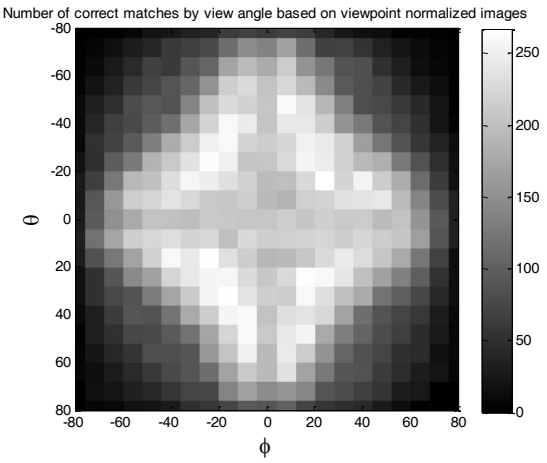

Figure 4. Number of correct feature matches by the viewpoint normalized OpenSURF as the function of viewpoint orientation (ϕ, θ : -80°..+80°)

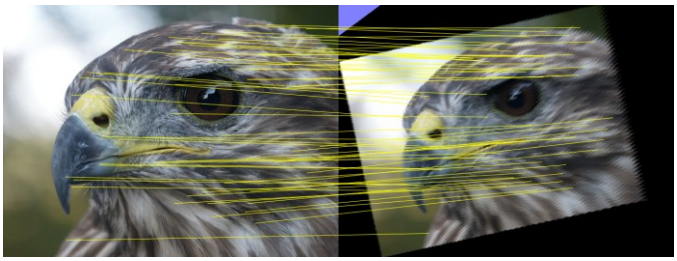

Figure 5. Matches between the original and the normalized image

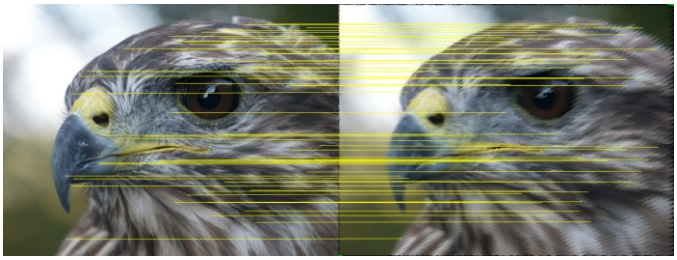

Figure 6. Correct matches between the original and the retransformed normalized image

The difference between the number of matches near (0°; 0°) (when the image is only slightly distorted) between the two algorithms is due to the sensitivity of SURF for scale. The normalized image was rendered at an arbitrary resolution. This affects the number of descriptors and matches.

Figure 7. shows the ratio of correct matches using the original SURF algorithm. Here the viewpoint change was only evaluated by rotating the image around one axis. At small viewpoint changes it produces reliable matches (low match threshold) and near 100% correct matches. As the viewpoint changes the threshold must be increased to produce any correct matches. At wider viewpoints and low thresholds no matches are found. As the match threshold is increased above 50° the ratio starts to fall suddenly.

Figure 8. shows the percent of correct matches when image viewpoint normalization is applied. Results show that the algorithm provides more stable feature detection. Correct matches are found at lower thresholds meaning stable and clear matches. The high ratio of correct matches are found at higher thresholds meaning that the descriptors are even more distinctive. Changes in viewpoint affects the match ratio less than in the previous case as it falls by a lower slope while still maintaining a higher ratio.

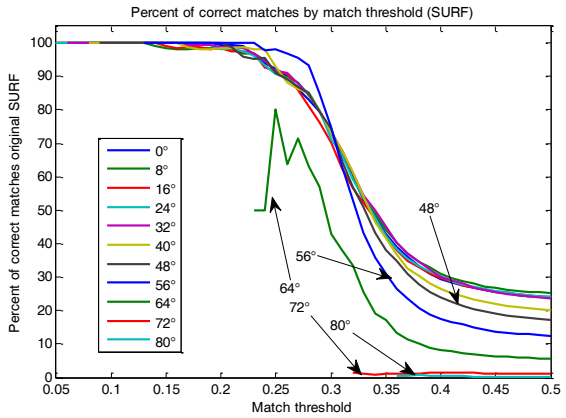

Figure 7. Ratio of correct matches as a function of match threshold using the original SURF algorithm

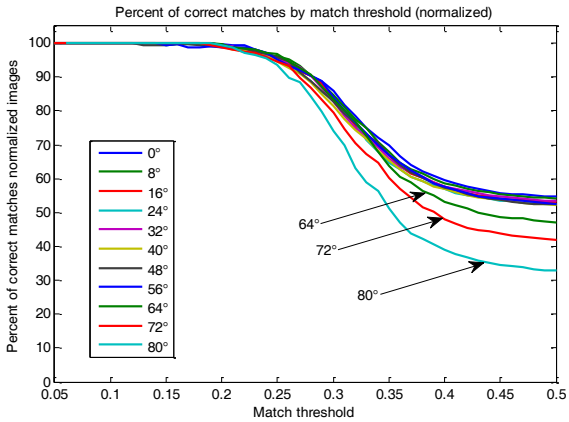

Figure 8. Ratio of correct matches as a function of match threshold using normalized images

B. *Results based on captured images*

During evaluation a printed picture was rotated. At each 15° step the matching features were searched using the original SURF and the modified using normalized images. Figure 9. shows a pair of captured images.

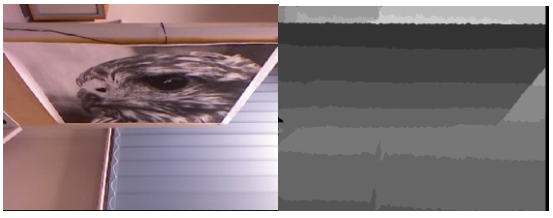

Figure 9. The captured intensity and range image

TABLE I. RESULTS BASED ON SENSOR DATA

Angle	-75°	-60°	-45°	-30°	-15°	+15°	+30°	+45°	+60°	+75°
Matches (original SURF)	0	6	57	124	156	158	112	26	0	0
Matches (normalized images)	31	106	98	146	160	149	148	107	57	5

Table 1. contains the results which show that at small viewpoint changes the normalizing algorithm has little benefit. But at larger viewpoint changes the method produces higher number of correct matches. The actual values depend on the image content. Matches were evaluated using 0.2 as threshold.

Figure 10. The viewpoint normalized image

V. CONCLUSION AND FUTURE WORK

In this paper we presented a method to improve matching capabilities of the SURF image feature extraction algorithm even at large viewpoint changes. The method utilizes the additional information gained from joined range and intensity images. We also showed simulation results and real world examples. Results show that it is worth utilizing additional range image data as it significantly increases the number and ratio of correct matches between images.

We look forward to continue the development to extend the algorithm to handle non-planar geometries more stable. Instead of the RANSAC more deterministic plane detection method algorithms will be evaluated. As RANSAC is based on randomly selecting samples from a set, different outcomes might appear in each run. We will evaluate different range image segmentation algorithms for this purpose. Gradient histogram based segmentation may be a good candidate algorithm. Future work also includes applying the algorithm to a SLAM implementation. The detection of feature points over a wider range of viewpoints are anticipated to improve the performance of the system.

ACKNOWLEDGMENT

The work reported in the paper has been developed in the framework of the project "Talent care and cultivation in the scientific workshops of BME" project. This project is supported by the grant TÁMOP - 4.2.2.B-10/1--2010-0009

REFERENCES

[1] A. E. Abdel-Hakim, and A. A. Farag, "CSIFT: A SIFT Descriptor with Color Invariant Characteristics", in *Proc. of International Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 1978-1983, 2006.

[2] C. Wu; B. Clipp; X Li; J.-M. Frahm, and M. Pollefeys, "3D model matching with Viewpoint-Invariant Patches (VIP)" in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp.1-8, Jun. 2008.

[3] D. Lowe, "Distinctive image features from scale-invariant keypoints", in *International Journal of Computer Vision*, vol. 60, pp. 91-110, 2004.

[4] D. M. Chu, and A. W. M. Smeulders, "Color Invariant SURF in Discriminative Object Tracking", *ECCV Workshop on Color and Reflectance in Imaging and Computer Vision*, 2010.

[5] G. J. Burghouts, and J.-M. Geusebroek, "Performance evaluation of local colour invariants", in *Computer Vision and Image Understanding*, vol. 113, issue 1, pp. 48-62, Jan. 2009.

[6] Guoshen Yu, and J.-M Morel, "A fully affine invariant image comparison method", in *Proc. IEEE International Conf. Acoustics, Speech and Signal Processing*, pp.1597-1600, Apr. 2009.

[7] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features", in *Computer Vision and Image Understanding*, vol. 110, issue 3 pp. 346-359, Jun. 2008.

[8] J. Bauer, N. Sünderhauf, and P. Protzel, "Comparing several implementations of two recently published feature detectors", in *Proc. of the International Conference on Intelligent and Autonomous Systems*, vol. 6. pt. 1. 2007.

[9] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "Evaluating Color Descriptors for Object and Scene Recognition", in *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.32, no.9, pp.1582-1596, Sept. 2010.

[10] K. Mikolajczyk, and C. Schmid, "A performance evaluation of local descriptors", in *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.27, no.10, pp.1615-1630, Oct. 2005.

[11] K. Mikolajczyk, and C. Schmid, "Scale & Affine Invariant Interest Point Detectors", in *International Journal of Computer Vision*, vol 60, issue 1, pp. 63-86, Oct. 2004

[12] Liang Cheng; Jianya Gong; Xiaoxia Yang; Chong Fan; and Peng Han; "Robust Affine Invariant Feature Extraction for Image Matching" in *IEEE Geoscience and Remote Sensing Letters*, vol. 5, no. 2, pp.246-250, April 2008.

[13] M. Ying Yang, Yanpeng Cao, and J. McDonald, "Fusion of camera images and laser scans for wide baseline 3D scene alignment in urban environments", in *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 6, pp. S52-S61, Dec. 2011.

[14] Y. Ke, and R. Sukthankar, "PCA-SIFT: a more distinctive representation for local image descriptors", in *Proceedings of International Conference on Computer Vision and Pattern Recognition*, vol.2, pp. 506-513, 2004

[15] Y. Cao, and J. McDonald, "Viewpoint invariant features from single images using 3D geometry" in *IEEE Workshop on Applications of Computer Vision*, pp.1-6, Dec. 2009.

[16] Y. Cao, M. Y. Yang, and J. McDonald, "Robust alignment of wide baseline terrestrial laser scans via 3D viewpoint normalization" in *IEEE Workshop on Applications of Computer Vision*, pp.455-462, Jan. 2011.