MVA2015 IAPR International Conference on Machine Vision Applications, May 18-22, 2015, Tokyo, JAPAN

Enhanced Surface Normal Computation by Exploiting RGB-D Sensory Information

Jens Hedrich
University of Koblenz-Landau
Universitätsstr. 1, 56070 Koblenz, Germany
jenshedrich@uni-koblenz.de

François Genois
Université de Sherbrooke
2500, Boulevard de l'Universit, Canada
francois.genois@usherbrooke.ca

Dietrich Paulus
University of Koblenz-Landau
Universitätsstr. 1, 56070 Koblenz, Germany
paulus@uni-koblenz.de

Marcin Grzegorzek
University of Siegen
Hoelderlinstr. 3, 57076 Siegen, Germany
marcin.grzegorzek@uni-siegen.de

Abstract

Reliable surface normal computation is fundamental for a broad range of computer vision application areas, e.g. object segmentation, classification and recognition. Naturally, the surface normal is computed on the acquired depth data, whereby the normal quality is dependent on noise performance and resolution of the underlying image modality. The tendency of combining different imaging sensors into one device is increasing and leads to a new sampling density, which can be used to compensate or reduce the drawbacks of modalities. This paper presents a novel method for computing surface normals on RGB-D images by combining superpixel segmentation in color space with plane fitting in depth space. For evaluation we perform a qualitative comparison between our method and two standard methods for computing normal maps in diverse indoor scenes. Our results show an improvement in computing normal maps with clear and crisp-edges. Furthermore, our proposed method approximates valid normal information in areas where the depth sensor returned errors or depth inhomogeneities. These results emphasize our assumption that under normal light conditions edges in depth space are coherent to edges in color space.

1 Introduction

Real time computing algorithms are important for a plethora of computer vision applications. The limitation in memory and computational power creates the need for efficient representations. State of the art algorithms in mobile robotics usually collect the perceived information of the environment in a 3-D point cloud. This representation is motivated by the presentability of the real world, as well as 3-D Laser Range Finder returning 3-D point sets. Generally, the noise characteristic is strongly varying between technologies such as 3-D Laser Range Finders, stereo cameras, Time-of-Flight and structured light sensors. Furthermore, combining different imaging sensors (e.g. color and depth) into one device results in more perceived information with the result that this data richness can be exploited for a better surface normal computation. However, depth sensors like Time-of-Flight or IR structured light sensors do not necessarily need to be converted into a point cloud structure, especially in applications where feature methods in the image space are used. Projecting of feature points back and forth between the cartesian image space and a point cloud can be computationally expensive. Computing normals in image space receives very little attention in literature, although noise performance of state of the art depth sensors would require comprehensive discussion.

The paper at hand is structured as follows: In section II we give an overview of related work in the field of surface normal computation. In section III we then introduce our method for enhanced surface normal computation. In section IV we describe our experiments and results to evaluate our approach. In section V we provide a final conclusions on our work and directions for future work.

2 Related Work

In recent years superpixel methods became increasingly popular in computer vision applications. Generally, superpixel methods can be broadly divided into graph-based and gradient ascent methods. Achanta et al. [1] empirically compared five state of the art superpixel algorithms. As a result the simple linear iterative clustering (SLIC) method outperformed the other superpixel algorithms in boundary adherence, segmentation speed, and performance. The segmentation speed of this method can be speeded up by a factor of 10x$\sim$20x with a GPU implementation [2].

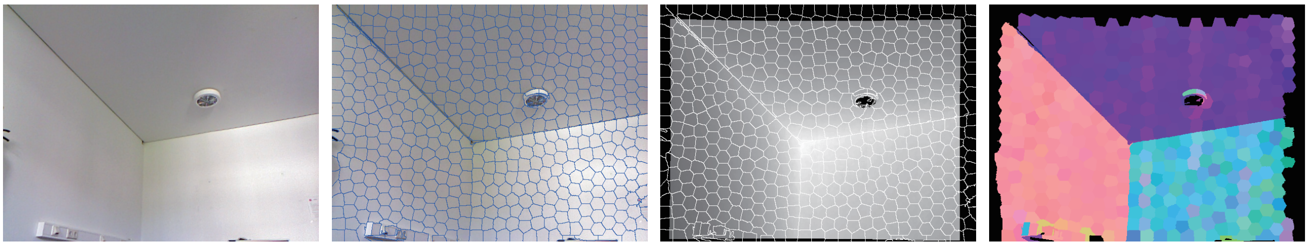
The surface normal estimation in point clouds has been increasingly investigated, while the normal estimation in the perceived depth image space has drawn little attention. Generally, the perceived data from stereo-, Time-of-Flight- or structured light cameras is projected into a point cloud.

Klasing et al. [3] analyzed and compared different methods for normal vector estimation in 3-D point clouds regarding the usability for online processing on mobile robots, with the conclusion that the presented *PlanePCA* is the universal method of choice.

Dey et al. [4] approximates the surface normals in point cloud data by the centers of a set of unique large Delaunay balls, named *polar balls*.

Badino et al. [5] proposes an approach for obtaining normals by calculating the derivatives of the surface from a spherical range image, in order to compute fast and accurate normals of a huge point cloud set.

Rasu describes in his PhD thesis [6] how surface normals can be estimated in a point cloud by searching for the k nearest neighbors in a kd-tree.



(a) (b) (c) (d)

Figure 1: (a) color image of a scene showing an upper corner of a room, (b) color image with mapped superpixel grid (c) depth image with mapped superpixel grid of the color image, (d) computed normal map of our proposed method.

Holzer et al. [7] proposes a method to estimate surface normals in integral images in real time under the usage of the covariance estimation.

Principal Component Analysis (PCA) is a statistical procedure to identify the principal directions in which the underlying data varies. These principal directions can be found by calculating the eigenvectors and eigenvalues of the data covariance matrix. This analysis is a standard procedure in computer vision applications. Neuhaus et al. [8] uses the principle component analysis to estimate the local flatness of outdoor 3-D point clouds, in order to distinguish between drivable and non-drivable areas in laser range data. Therefore, the ratio between the three eigenvalues of the hierarchically subdivided point cloud is analyzed.

Diebel et al. [9] successfully combines a high resolution color image with a lower resolution Time-of-Flight-sensor to increase the low resolution of the depth. Therefore, Markov Random Fields are used to exploit the tendency that discontinuities in range and coloring co-align.

3 Enhanced Surface Normal Computation

In this section we describe our simple but effective method for surface normal computation on RGB-D images. The presented method requires registered color and depth information of a scene. Further, the depth information needs to be encoded in a metric 3-D distance space. Depth inhomogeneities or errors, which can appear through the nature of a Time-of-Flight or a structured light depth sensor (e.g. Microsoft Kinect V1), can be handled by our method and do not need any preprocessing. Once the color and depth data is fused we compute the normal map in three major steps: 1) Superpixelation of the RGB data. With this method we cluster our scene into equally sized superpixel tiles. The superpixel segmentation method preserves sharp boundaries in the color image. 2) Normal computation. We use the superpixel grid for clustering the depth information in compact tiles, because we assume that under normal illumination conditions an edge in 3-D space generates a visible boundary in color space (cf. Diebel et al. [9]). This methodology enables us to compute normals in depth image space, leading to the advantage that we do not need to organize the depth points in a special point cloud structure, such as kd-tree. Within each superpixel tile we randomly select depth points to compute the locally approximated normal information by fitting a plane model. Therefore, we use covariance matrix for computing the eigenvalues and RANSAC to neglect outliers. 3) Finally, we generate the normal map by assigning the estimated normal to all pixels of the superpixel. Figure 1 (a-d) shows an example to illustrate the main steps of our method.

3.1 Superpixelation

To superpixelate the color image, we use the SLIC method proposed by [10]. The main advantage of this superpixel segmentation is that its results in equally sized compact superpixel tiles. The parameters: size of the superpixel and the regularization factor define the nature of our normal maps. Small superpixels are fast to compute and lead to more details, although local noise will remain within the normal map. On the other hand, bigger superpixels are perfect for reducing the noise within its closed range on planes, e.g. a wall.

3.2 Normal Computation

For estimating the normal we compute the eigenvalues of the covariance matrix of the 3-D points within each superpixel. The lowest eigenvalue corresponds to the approximated normal of the point set. The point distribution of the set lies on a plane if the other two eigenvalues are significantly higher compared to the lowest eigenvalue. Figure 2 (a) shows the depth information of a door detail with the overlaid superpixel grid of the color image. It can be recognized that the depth information at the superpixel-border includes the majority of outliers. These outliers change the normal orientation in areas where depths values are strongly varying. In Figure 2 (b) we illustrate the effect when the normal is computed based on all depth points within one superpixel and Figure 2 (c) is an illustration for computing the normal based on all depth points of the superpixel but without the border pixels. In our method we use RANSAC to generate a number of random k plane candidates. Therewith, we reduce the computational complexity and ensure that outliers are neglected. As described above, the points for the plane model are randomly picked from depth points within one superpixel without the border pixels. The plane model with the highest consensus is selected for computing the normal of the superpixel. We calculate the required number of candidates for an expected number of outliers with a desired precision based on the RANSAC termination criteria $k = log(1-p)/log(1-w^n)$, where p is the precision, w the number of inliers and n the number of points for estimating


(a) (b) (c) (d)

Figure 2: image detail of Figure 3a (red rectangle), (a) depth image with mapped superpixel grid of the color image, note the depth outlier within the superpixel grid, (b) result of our method by selecting all points for normal estimation, (c) result by selecting all points but the border of a superpixel, (d) result by selecting random points of a superpixel without the border points.

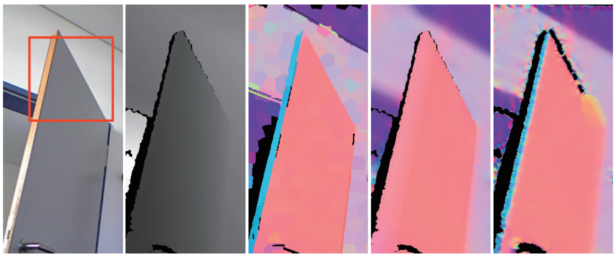

(a) (b) (c) (d) (e)

Figure 3: Office door: (a) color image, (b) depth image, (c) result of our proposed method with a superpixel size of 20 and a regularization factor of 0.3, (d) result of method [6] with a kd-tree radius search of 10cm, (e) result of method [7] with the parameters of a normal smoothing size of 30 and a max depth change factor of 0.2.

the model. Figure 2 (d) shows the improvement of calculating the normal of each superpixel.

4 Experiments & Results

In this section we present the qualitative results of our method. Further, we compare our work with the normal estimation methods proposed by [6] and [7]. The parameters for our method are mainly determined by the size and regularization factor of the superpixel algorithm. The threshold and termination criteria of RANSAC are fixed values in our experiments. As threshold distance we used 2.5cm. This threshold is motivated by the average depth resolution along the z-axis of the used depth sensor. Moreover, we assume an inliers ratio of $w = 0.6$ and a precision of $p = 0.98$, which lead to termination criteria of $k = 16$ plane candidates for each superpixel.

We evaluated our method in three scenarios where we captured an untidy office, doors and a human. Figure 3 shows a comparison between the point cloud library implementations of methods [6], [7] and our method showing a door. It can be recognized that our method performs best on flat objects in the real world. The advantage of our method is illustrated in Figure 3 (c). Due to the nature of superpixels we are able to generate normal information in areas where no depth information was captured. Additionally, it should be noted that the thin door edge is covered with the correct normal orientation. The method proposed by [6] strongly smoothes the scene and [7] increases the shadow area.

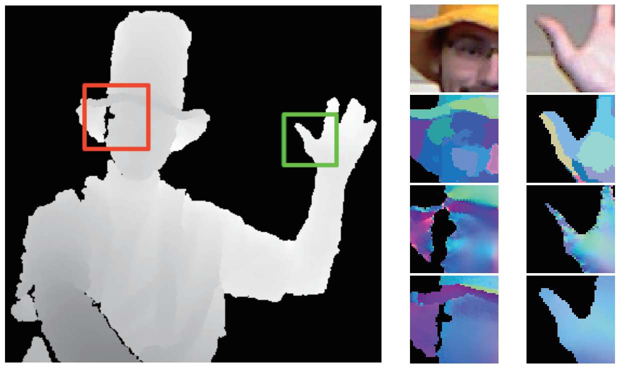

(a) (b) (c)

Figure 4: (a) depth image of a person, (b-c) comparison of problematic areas, the fist row shows the color information, the second row depicts the result of our method with a superpixel size of 15 and a regularization factor of 0.8, the third row shows the result of [7] with a normal smoothing size of 20 and a max depth change factor of 0.2, the bottom row shows the result of [6] with a kd-tree radius search of 5cm.

In Figure 4 we compare the methods with selected details of a scene showing a human. The face detail in Figure 4 (b) shows that normals are well approximated even for natural shapes. The hand detail in Figure 4 (c) reveals that with our proposed method the full hand region is covered within the normal map, while the other methods reduce the size of the hand.

Our final test scenario in Figure 5 shows an untidy office with round and flat objects in the near (< 2m) and far space. Figure 5 (c) and (d) illustrate the effect and importance of the superpixel size. Small superpixels need to be used for small and round objects in near space. Bigger superpixel size consequently reduces details of objects in the scene, while huge homogenous planes such as the ground floor or walls can be easily covered.

The runtime of our proposed method is strongly dependent on the superpixel size. Clustering the color image into bigger superpixel increases the computational complexity. Within our current implementation the SLIC superpixel method consumes around 2/3 of the computational time. Hence, a normal map with a superpixel size of 8 pixels is computed approximately in 0.8 seconds. However, this could be enhanced with a GPU base SLIC implementation.

5 Conclusion

In this paper we presented a simple but novel method to compute a normal map in the image space by exploiting the RGB-D-sensory information. The quality of the normal map is still depending on the noise characteristics of the underlying imaging technology. We showed that superpixels can be used for clustering the color image in logically connected tiles

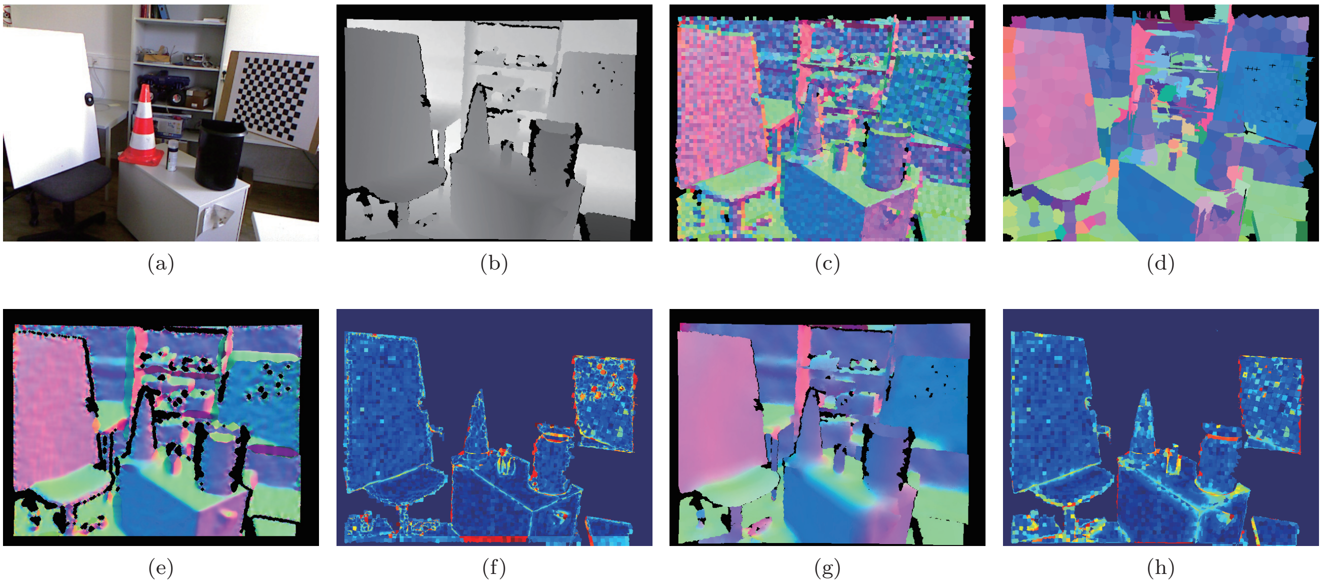


Figure 5: (a) color image, (b) depth image, (c) our method with a superpixel size of 8 and regularization factor of 0.8, (d) our method with a superpixel size of 25 and regularization factor of 0.3, (e) [7] with a normal smoothing size of 20 and a max depth change factor of 0.2, (f) difference map of the near space (0.2m - 2.1m) of Figure 5 c and e, (g) [6] with a radius search of 10cm, (h) difference map between Figure 5 c and g.

and the supplemented depth information for computing the normals in this region. However, this method is strongly dependent on the light condition of the scene. The quality of our normal estimation is evident under normal light condition. The method preserves the true shape of objects without blurring or shifting the object edges. The only drawback is that round objects are difficult to approximate by superpixel tiles. For such object types the local normal error is increased. As a solution the ratio of the three eigenvalues could be investigated in a future work to predict the roundness of each superpixel. Besides, improvements are planned to use a GPU variant of the SLIC algorithm in order to make our method computable in real time. Additionally, we plan to use this method in conjunction with DP-Means-Clustering for RGB-D image segmentation.

Acknowledgments

This work was funded by the German Research Foundation (DFG) as part of the research training group GRK 1564 'Imaging New Modalities'.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, pp. 2274–2282, Nov 2012.

[2] C. Y. Ren and I. Reid, "gslic: a real-time implementation of slic superpixel segmentation," tech. rep., University of Oxford, Department of Engineering Science, 2011.

[3] K. Klasing, D. Althoff, D. Wollherr, and M. Buss, "Comparison of surface normal estimation methods for range sensing applications," in *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pp. 3206–3211, May 2009.

[4] T. Dey and J. Sun, "Normal and feature approximations from noisy point clouds," in *FSTTCS 2006: Foundations of Software Technology and Theoretical Computer Science* (S. Arun-Kumar and N. Garg, eds.), vol. 4337 of *Lecture Notes in Computer Science*, pp. 21–32, Springer Berlin Heidelberg, 2006.

[5] H. Badino, D. Huber, Y. Park, and T. Kanade, "Fast and accurate computation of surface normals from range images," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 3084–3091, May 2011.

[6] R. B. Rusu, *Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments*. PhD thesis, Computer Science department, Technische Universitaet Muenchen, Germany, October 2009.

[7] S. Holzer, R. Rusu, M. Dixon, S. Gedikli, and N. Navab, "Adaptive neighborhood selection for real-time surface normal estimation from organized point cloud data using integral images," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 2684–2689, 2012.

[8] F. Neuhaus, D. Dillenberger, J. Pellenz, and D. Paulus, "Terrain drivability analysis in 3d laser range data for autonomous robot navigation in unstructured environments," in *Emerging Technologies Factory Automation, 2009. ETFA 2009. IEEE Conference on*, pp. 1–4, Sept 2009.

[9] J. Diebel and S. Thrun, "An application of markov random fields to range sensing," in *Proceedings of Conference on Neural Information Processing Systems (NIPS)*, (Cambridge, MA), MIT Press, 2005.

[10] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, pp. 2274–2282, Nov 2012.