IMPLICIT SCENE MODELLING FROM IMPRECISE POINT CLOUDS

E. Funk [a,b], L. S. Dooley[a], A. Boerner[b], D. Griessbach[b]

[a] Department of Computing and Communications
The Open University
Milton Keynes, United Kingdom
laurence.dooley@open.ac.uk

[b] Department of Optical Information Systems
Institute of Robotics and Mechatronics
DLR (German Aerospace Center), Berlin, Germany
(eugen.funk, anko.boerner, denis.griessbach)@dlr.de

Commission IV/7

KEY WORDS: Implicit Surface, General Lasso, Sparse Approximation

ABSTRACT:

In applying optical methods for automated 3D indoor modelling, the 3D reconstruction of objects and surfaces is very sensitive to both lighting conditions and the observed surface properties, which ultimately compromise the utility of the acquired 3D point clouds. This paper presents a robust scene reconstruction method which is predicated upon the observation that most objects contain only a small set of primitives. The approach combines sparse approximation techniques from the *compressive sensing* domain with surface rendering approaches from computer graphics. The amalgamation of these techniques allows a scene to be represented by a small set of geometric primitives and to generate perceptually appealing results. The resulting scene surface models are defined as implicit functions and may be processed using conventional rendering algorithms such as marching cubes, to deliver polygonal models of arbitrary resolution. It will also be shown that 3D point clouds with outliers, strong noise and varying sampling density can be reliably processed without manual intervention.

1 INTRODUCTION

While many low cost 3D sensors such as stereo cameras and Kinect have recently become available, they all have difficulties in providing reliable object and surface reconstruction under either varying lighting conditions or during motion as for example, in mobile systems. There is thus a need for robust algorithms to be developed which are able to extract semantic and geometric information from the imprecise point clouds. Existing approaches including (Ohtake et al., 2003, Kazhdan et al., 2013) and (Alexa et al., 2001) from the computer graphics domain, are able to process data which has either low noise or contains a small number of outliers. This is because they have been designed to deliver visually appealing renderings, so the semantic structure of the scene is not considered. Information concerning the geometrical properties of a scene or of object is important for perception applications where semantic information is required for scene recognition and navigation.

Being interested in the underlying geometrical scene structure, (Schnabel et al., 2007) performed an explicit search for a predefined set of expected primitives by applying the RANSAC (Fischler and Bolles, 1981) method. The detected primitives however, are difficult to use for visual rendering. The scenes contain many holes since the distribution of the detected elements is very sparse in comparison with points employed in computer graphics methods. In order to overcome this issue (Lin et al., 2013) used groups of detected primitives to recognize an object type and searched a database for the most similar manually created 3D model, which was then employed for rendering. In this paper a connection between sparse primitive-based scene modelling and dense rendering is established with the introduction of a new approach to extract planar primitives from the point clouds by describing surfaces by implicit functions. This allows dense rendering and the filling in of holes caused by missing samples. Additionally, the number of primitives used to parametrize a scene is penalized in the convex optimization process which leads to very sparse, but crucially smooth representation model. The optimization algorithms are designed to identify and remove both noise and outliers from the input data, thereby increasing their robustness.

The remainder of this paper is structured as follows: Section 2 reviews shortly traditional methods for surface reconstruction from the machine learning domain, where nonlinear classification models are solved by sparsity inducing linear optimization. Section 3 then presents the data processing framework and sparse scene approximation method, with a corresponding results discussion on the method's performance being provided in Section 4. Section 5 describes some possible future research directions and makes some concluding comments.

2 SURFACE RECONSTRUCTION

This section reviews some of the common methods for surface reconstruction and the established class of surface modeling algorithms which directly create triangle-based meshes between 3D points, known as Delaunay triangulation (Su and Drysdale, 1995). Interpolating via triangulation on noisy data is inappropriate however, because already small measurement errors directly degrade the surface accuracy. To solve this problem combining the recognition of either edges or other simple features in 3D point clouds with regression methods has been proposed by (Lempitsky and Boykov, 2007) using a global optimization technique called graph cuts (Boykov et al., 2001). Global approaches are however, computational expensive and cannot be applied to large point clouds. An alternative class of surface reconstruction

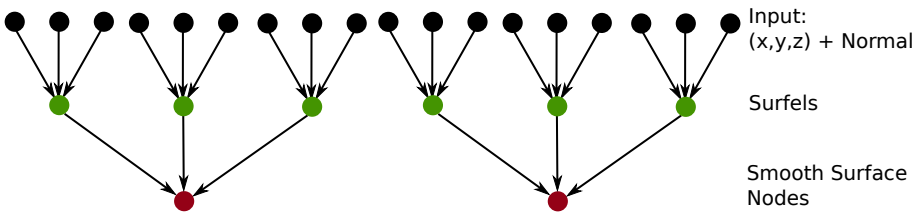


Figure 1: The hierarchical processing architecture. The inputs (3D points + normals) in a local neighbourhood are used to estimate the parameters of a larger planar patch, the surfels, which model the implicit function.

methods relies on the search of an implicit function, which defines the surface of interest in zero space, as for example in the sphere equation $x^2 + y^2 + z^2 - r^2 = 0$, where the 3D space is defined over x, y, z. Implicit models for surface reconstruction have been proposed by (Gomes et al., 2009, Ohtake et al., 2003, Oztireli et al., 2009) which enable smoothing and effective extrapolation for hole filling in the scanned samples. However, these methods are also sensitive to noise and the optimization task is nonlinear so they can only be solved in small local domains. Being motivated by the robust generalization properties and capability to solve nonlinear problems by linear regression, global kernel based regression models have been proposed by (Walder et al., 2007). Their main benefits are the convex optimization framework, linearity of target parameters and the potential to apply regularization to enforce smoothness or a priori knowledge about the data structure, e.g. predictable vertical or horizontal surfaces. Implicit kernel based models have been studied by (Duchon, 1977, Wahba, 1990), where the application of different kernel types is discussed. Extending the linear kernel regression with sparsity inducing regularization allows surface reconstruction and the selection of appropriate surface model in one convex optimization step. The proposed framework utilizes these elements and integrates the linear surface reconstruction along with robust l_1 regularization, known from the compressive sensing discipline. Specifically, l_1 norm is also known to be less sensitive to noise and outliers, which has been demonstrated in (Candès and Randall, 2006). The test dataset incorporates the 3D acquisition results from (Griessbach et al., 2010), with a moving stereo system providing accurate camera trajectory plus dense stereo images. Similar to hierarchical learning strategies, enabling stepwise efficient regression, the surface reconstruction process is divided into two steps which are now described.

3 THE PROPOSED METHOD

The approach is hierarchical with the first input layer defined by 3D points with normals, and the results being subsequently input to the surface representing function (figure 1). If no normal vectors in raw 3D points are available, the efficient *principal component analysis* (PCA) method (Pearson, 1901, Klasing et al., 2009) can be applied. Unfortunately, 3D reconstruction of indoor or outdoor scenes inherently contains non-uniform noise, so the Gaussian error model is not applicable. This constraint means that the 3D points and corresponding normals need to be corrected by a further process. The data is structured with an OcTree (Wurm et al., 2010), which enables efficient access to a points' neighbours and thus hierarchical processing. The first step uses the 3D points from a local neighbourhood to reconstruct local planar patches (surfels) using l_1 regularization to reduce the effects of noise. This step is illustrated by the first level in figure 1. The local planar patches are then used in the second processing step where the implicit function is estimated by applying kernel based regression and the l_1 robust statistics. The implicit function is defined for a larger but still local domain and can be estimated efficiently. The introduction to regularized l_1 regression will now be considered.

3.1 Introduction to Sparse Regression

The presented method has its foundations in the compressive sensing domain and theories on sparse signal reconstruction. The technique pragmatically assumes that linear regression using fewer measurements than target variables is feasible when the cost function is augmented by a regularization term (Tikhonov, 1943). This penalizes the model complexity, utilizes domain knowledge and helps to avoid overfitting even in under-sampled problems. Recently, (Candès and Randall, 2006) presented a technique utilizing l_1 regression for error correction. The usual *least squares* (LSQ) problem $\| y - Ax \|$ is augmented by error vectors e to $\| y - Ax - e \|$ which are also required to be estimated. Moreover, the level of error to be penalized is given by $\lambda \| e \|_1$, where λ is the weight of the penalizing term. Figure 2 demonstrates the case, where parameters of a simple line are estimated from samples with strong non-uniform noise. In this example, the model of interest is $y = mx + d$ with m as the slope and d the offset. The parameters $x = (m, d) \in \mathbb{R}^2$ are the model target variables which are estimated jointly with $\mathbf{e}$ in

$$\| y - A\mathbf{x} - \mathbf{e} \|_2 + \lambda \| \mathbf{e} \|_1 . \quad (1)$$

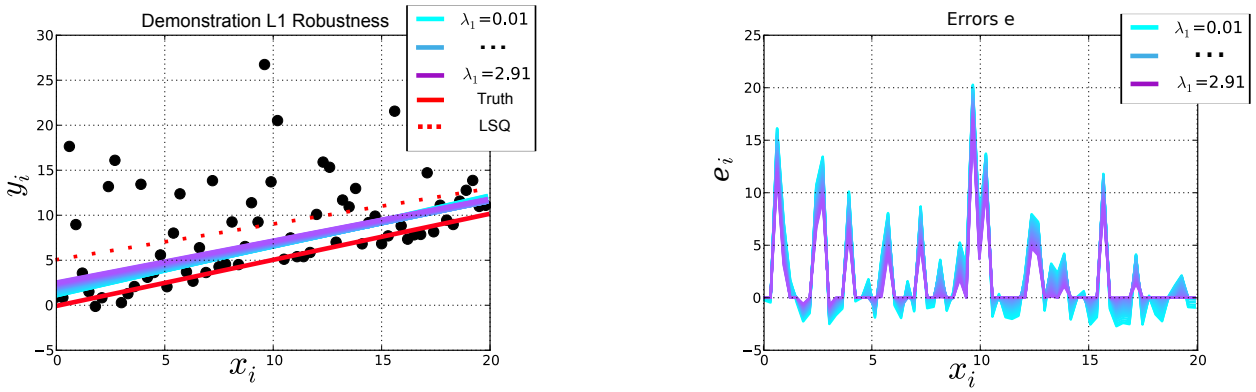
Since the error vector $\mathbf{e}$ is likely to be sparse, the l_1 norm $\| \mathbf{e} \|_1$ provides a convenient choice. In the following sections this observation is utilized for two particular steps:
i) The plane parameters of the local surfels are estimated using the error slack variables approach.
ii) In reconstructing the overall implicit surface function from the set of previously computed surfles.

The number of surfels used to describe the higher level surface function is penalized by l_1 because it is anticipate to use only a few surfels to represent the overall surface. The l_1 norm is not differentiable so it is more difficult to solve than LSQ problems. Nevertheless, the objective function is still convex and can be solved efficiently with existing approaches (Goldfarb and Idnani, 1983, Afonso et al., 2010) as demonstrated in the following sections.

3.2 Surfel Regression

The planar model of a surfel is defined as a simple linear plane $f(\mathbf{x}) = b(\mathbf{x})^T \mathbf{c}$, with $b(\mathbf{x}) = (x_1, x_2, x_3, 1)$ as the first degree polynomial and $\mathbf{c}$ as the parameters of interest. $\mathbf{c} = (\mathbf{n}, d)$ contains the normal vector $\mathbf{n}$ of the plane and its offset d, so that $b(\mathbf{x})^T \mathbf{c} = \mathbf{n}^T \mathbf{x} + d$ corresponds to a plane equation. Similar to the l_1 regression in Section 3.1, the surfel regression task considers the errors as a residuum vector $\mathbf{e}$ as a part of the target variables to be estimated:

$$\min_{\mathbf{c},\mathbf{e}} \sum_{i}^{N} \| \mathbf{n}_i - \nabla f(\mathbf{x}_i) \|_2 + \lambda_1 \| f(\mathbf{x}_i) + e_i \|_1 + \lambda_2 \| e_i \|_1 \quad (2)$$

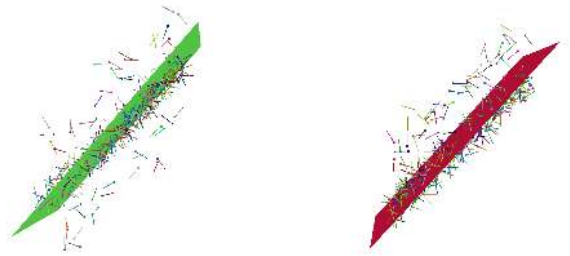


a) b)

Figure 2: Robust l_1 regression, 20 experiments with varying λ. Compared to the l_2 regression (LSQ, red dotted line) the estimated parameters are closer to the ground-truth (bottom solid red line). b) The error value e_i for every sample x_i. Increasing the weighting λ on $\| e \|_1$ increases its sparsity.

where $\mathbf{n}_i$ is normal vector of the 3D point sample $\mathbf{x}_i$, and e_i is the corresponding error slack-variable. The weighting factors λ_1 and λ_2 control the strength of the penalization relative to the $\| \cdot \|_2$ part, which are usually estimated via cross validation.

Figure 3 illustrates the regression behaviour of the augmented l_1 regression with synthetic random normals, and with uniform and non-uniform noise, together with outliers. Despite the presence of strong non-uniform noise and outliers, the plane has been recovered satisfactorily. The integration of the error variables e plays also here the key role in eliminating outliers. Since no noise distribution is modelled and the regularization of $\| e \|_1$ penalizes only the deviation from the estimated model, the noise and outliers are not expected to follow Gaussian Normal distribution.



a) b)

Figure 3: Demonstration of the plane regression with l_1 regularization and the augmented error slack variables. In both cases a) and in b) the normal vectors to every point are random. The noise in the point positions is a) uniformly and b) non-uniformly distributed.

3.3 Additive Implicit Functions

At the higher level of processing hierarchy, surfels are the input and are incorporated into a function $f(\mathbf{x})$. Similar to the previously introduced plane equation, f defines the surface of interest s implicitly by the zero set of f:

$$s = \{\mathbf{x} : f(\mathbf{x}) = 0\}, \mathbf{x} \in \mathbb{R}^3.$$

which means that all 3D points $\mathbf{x}$, for which f returns zero, are on the surface. Moreover, $f(\mathbf{x})$ is defined as a linear combination of N basis functions $f_j(\mathbf{x})$ plus a polynomial $b(\mathbf{x})^T\boldsymbol{\beta}$ of 1st degree

$$f(\mathbf{x}) = \sum_j^N f_j(\mathbf{x})\alpha_j + b(\mathbf{x})^T\boldsymbol{\beta} \quad \text{with } b(\mathbf{x}) = (x_x, x_y, x_z, 1) \tag{3}$$

where N is the number of input surfels and f_j is defined by the j-th input surfel. The model is based on additive and regularized splines (Duchon, 1977, Wahba, 1990) which has been successfully applied in different disciplines including machine learning and classification, where the basis functions f_j are known as *kernels* and the factors α_j the *support vectors* (Scholkopf and Smola, 2001, Bishop, 2007). Here, the basis functions incorporate the previously estimated surfel parameters and are defined as

$$f_j(\mathbf{x}) = r^2[(\mathbf{x}_j - \mathbf{x})^T\mathbf{n}_j]^2 \tag{4}$$

where $r = \| \mathbf{x}_j - \mathbf{x} \|_2^2$, $\mathbf{x}_j$ is the center of the i-th surfel plane and $\mathbf{n}_j$ is its normal vector. The kernel f_j enhances planar surface models since for every point on the surfel-plane f_j returns zero.

When rewriting (3) in matrix form, the regression task for f is defined as

$$\| \mathbf{n} - \begin{pmatrix} K_\nabla & B_\nabla \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix} \|_2 + \| \begin{pmatrix} K & B \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix} \|_1 + \lambda_a \| \boldsymbol{\alpha} \|_1 \tag{5}$$

where the $\| \cdot \|_2$ term performs regression on the surface orientations and the first $\| \cdot \|_1$ term penalizes the surface function f when it does not return zero at the position of one of the sample points $\mathbf{x}_i$. The final l_1 part $\| \boldsymbol{\alpha} \|_1$ penalizes the number of models used to form f and crucially reduces the complexity of the representation. The vector $\mathbf{n} \in \mathbb{R}^{3 \cdot N}$ contains the stacked normal vectors from the *input PCA-points*, K is known as the *kernel matrix* in the machine learning community. Here it is defined as $K_{m,j} = f_j(x_m)$. Similarly, the j-th row of B is $b(\mathbf{x}_j)^T$ as it is defined in (5). The gradient $\nabla f(\mathbf{x}_i)$ is expected to be equal to the measurements of the surface normals $\mathbf{n}_i$, which is considered by K_∇ and B_∇ in the l_2 part. In contrast to the function model used (3) it is common to use Gaussian radial basis functions for f_j to build the kernel matrix. The main reason for this is that Gaussian kernels provide a positive definite kernel matrix K, which ensures the regression task (5) is solvable. However, the calculation of exponential functions is very time consuming and is avoided in this proposed method. Unfortunately, positive definite K cannot be achieved with the proposed kernel in (4) as the sample points

are located in the zero space of f_j, which induces a positive semi definite (PSD) K. From (Duchon, 1977) and (Wahba, 1990) it is known that when PSD basis functions are employed, a polynomial $b(\mathbf{x})^T\boldsymbol{\beta}$ must be added to the linear combination $\sum_i f_j\alpha_j$ to make the linear system solvable. However, the polynomial coefficients must be orthogonal to the support vectors $\boldsymbol{\alpha}$, which results in a constrained optimization task. This can be expressed in matrix notation as $B(\mathbf{x})^T\boldsymbol{\alpha} = \mathbf{0}$ where the m-th column of B^T is $b(\mathbf{x}_m)$.
Moreover, similar to the surfels estimation process, the regression of the high level function f is augmented by the error slack variables $\mathbf{e}$ so the final regression thus becomes:

$$\min_{\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{e}} \| \begin{pmatrix}\mathbf{n}\\ \mathbf{0}\end{pmatrix} - \begin{pmatrix}K_\nabla & B_\nabla & \mathbf{0}\\ B^T & \mathbf{0} & \mathbf{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\beta}\\ \mathbf{e}\end{pmatrix} \|_2 + \lambda_0 \| \begin{pmatrix}K & B & I\end{pmatrix}\begin{pmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\beta}\\ \mathbf{e}\end{pmatrix} \|_1 + \lambda_e \| \mathbf{e} \|_1 + \lambda_a \| \boldsymbol{\alpha} \|_1 . \quad (6)$$

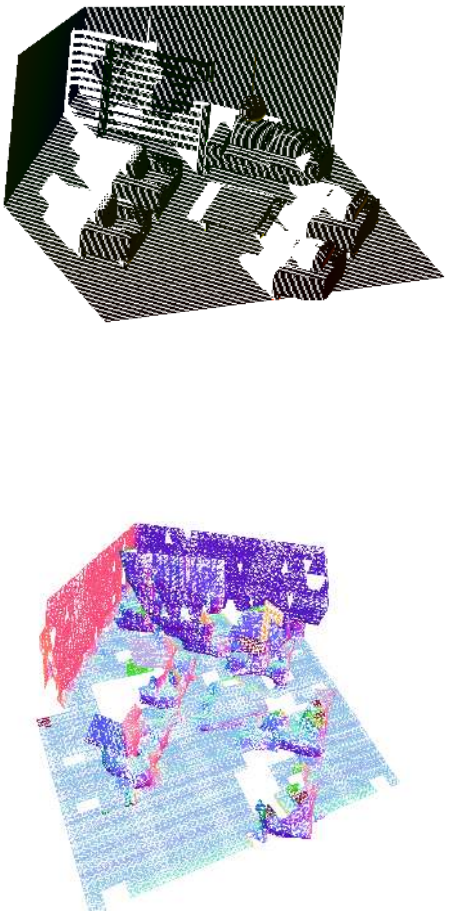
4 RESULTS

The first set of reconstruction experiments were performed upon a synthetic scene, where the acquisition of the 3D point cloud has been simulated using *Blender* (Gschwandtner et al., 2011), which renders a synthetic geometric model as if it is viewed by a depth camera. Figure 4 shows the synthetic point cloud and the corresponding reconstruction results. The upper part of 4 a) shows the point synthetic point cloud, where the severe holes caused by occlusions are clearly visible. Missing 3D samples are very common in real data sets, thus the experiment results in 4 b) illustrate the capability of the presented method to close the holes. The parameters for all experiments have been set to $\lambda_0 = 10$, $\lambda_a = \lambda_e = 0.1$. The reconstructed implicit function has been rendered by the *marching cubes* (Lorensen and Cline, 1987) method, which basically divides the 3D space into a regular grid and the function $f(\mathbf{x})$ computed on every grid node $\mathbf{x}$ in order to generate polygons suitable for standard geometry visualization software. For reasons of efficiency, the rendering has been applied on every f-node individually, where each top level surface function f covers a local domain of approximately $3m$. Many holes have been filled and the overall structure of the scene has been successfully recovered, though there are some new small holes in areas not covered by the marching cubes rendering process. This is because the function domains do not overlap in the underlying OcTree space splitting process.

In the second experiment a larger environment has been scanned by a passive mobile stereo camera (Griessbach et al., 2010), with the aim to recover the environment while the camera has been moving at walking speed inside a building. The raw point cloud is shown at the top of figure 5, with the corresponding reconstruction results being displayed in Figures 5 a) and b). Despite of high movement speed, reflecting floor and luminous wall decorations, the structure of the corridor has been recovered successfully.

5 OUTLOOK AND CONCLUSION

The surface is modelled piecewise in hierarchical manner, which enables piecewise smooth and efficient yet not competing results with high cost 3D sensors. The kernel function models and the rendering techniques will be in the focus of further research, which is crucial for the surface properties and the hole filling capabilities. Also the specific domain knowledge, such as similarities in the colour image space will be integrated since the applied kernel



a)

b)

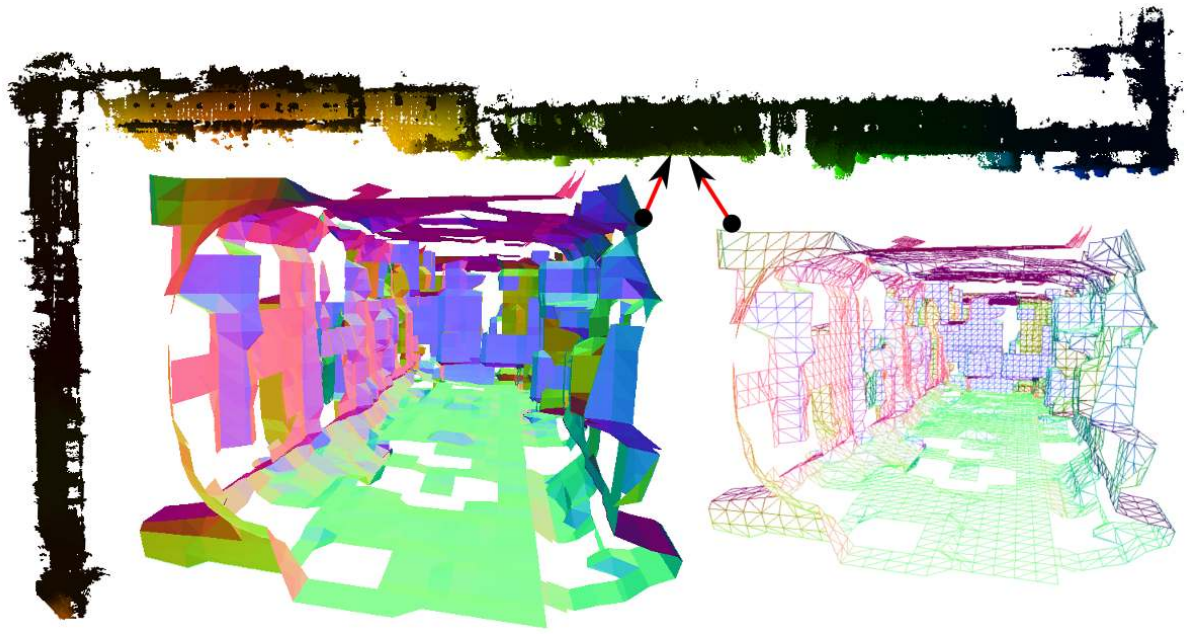
Figure 4: Reconstruction of a synthetic scene with no noise but large occluded areas. a) Synthetically generated 3D points. b) The reconstruction results, where the colour shading represents the normal orientation of the new surface.

based approach enables flexible extensions for a priori information. Motivated by the need to perform semantic and geometrical reconstruction of 3D point clouds a method closing this bridge has been developed. The framework exploits implicit surface models on coarse and imprecise 3D point data and allows to close occlusion holes and to create polygon meshes of arbitrary resolution. Extending the implicit surface models with the sparsity inducing l_1 regularization enables to penalize the reconstruction model complexity and to remove non uniform noise and outliers efficiently. The produced results are suitable for rendering and for semantic analysis of the underlying scene, which is a novelty today. Future work will cover the research on suitable function kernels and the numerical approaches to obtain sparse solutions.

REFERENCES

Afonso, M., Bioucas-Dias, J. and Figueiredo, M. A. T., 2010. Fast image recovery using variable splitting and constrained optimization. Image Processing, IEEE Transactions on 19(9), pp. 2345–2356.

Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D. and Silva, C. T., 2001. Point set surfaces. In: Proceedings of the conference on Visualization '01, VIS '01, IEEE Computer Society, Washington, DC, USA, pp. 21–28.



a) b) c)

Figure 5: Reconstruction results from a part of the large corridor dataset. The part is marked with two red lines on the top. The proposed method recovered the structure of the corridor successfully despite of reflecting floor and overexposed wall decorations. a) Top view of the raw point cloud which has been acquired by a mobile depth sensor. b) Triangulated result rendered with non transparent meshes. c) Wire frame view of the same part of the corridor.

Bishop, C. M., 2007. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer.

Boykov, Y., Veksler, O. and Zabih, R., 2001. Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 23, pp. 2001.

Candès, E. J. and Randall, P. A., 2006. Highly robust error correction by convex programming. CoRR.

Duchon, J., 1977. Splines minimizing rotation-invariant seminorms in sobolev spaces. In: W. Schempp and K. Zeller (eds), Constructive Theory of Functions of Several Variables, Lecture Notes in Mathematics, Vol. 571, Springer Berlin Heidelberg, pp. 85–100.

Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24(6), pp. 381–395.

Goldfarb, D. and Idnani, A., 1983. A numerically stable dual method for solving strictly convex quadratic programs. Mathematical Programming 27(1), pp. 1–33.

Gomes, A., Voiculescu, I., Jorge, J., Wyvill, B. and Galbraith, C., 2009. Implicit Curves and Surfaces: Mathematics, Data Structures and Algorithms. 1st edn, Springer Publishing Company, Incorporated.

Griessbach, D., Börner, A., Ernst, I. and Zuev, S., 2010. Real-time dense stereo mapping for multi-sensor navigation. In: International Society for Photogrammetry and Remote Sensing (ISPRS), Vol. 38, pp. 256–261.

Gschwandtner, M., Kwitt, R., Uhl, A. and Pree, W., 2011. BlenSor: Blender Sensor Simulation Toolbox Advances in Visual Computing. Lecture Notes in Computer Science, Vol. 6939, Springer Berlin / Heidelberg, Berlin, Heidelberg, chapter 20, pp. 199–208.

Kazhdan, M., Hopkins, J. and Hoppe, H., 2013. Screened poisson surface reconstruction. Transactions on Graphics (To Appear).

Klasing, K., Althoff, D., Wollherr, D. and Buss, M., 2009. Comparison of surface normal estimation methods for range sensing applications. International Conference on Robotics and Automation.

Lempitsky, V. and Boykov, Y., 2007. Global Optimization for Shape Fitting. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Minneapolis, USA.

Lin, H., Gao, J., Zhou, Y., Lu, G., Ye, M., Zhang, C., Liu, L. and Yang, R., 2013. Semantic decomposition and reconstruction of residential scenes from lidar data. ACM Transactions on Graphics, (Proc. of SIGGRAPH 2013).

Lorensen, W. E. and Cline, H. E., 1987. Marching cubes: A high resolution 3d surface construction algorithm. SIGGRAPH Comput. Graph. 21(4), pp. 163–169.

Ohtake, Y., Belyaev, A., Alexa, M., Turk, G., Seidel, H.-P. and Saarbrcken, M., 2003. Multi-level partition of unity implicits. ACM Transactions on Graphics 22, pp. 463–470.

Oztireli, C., Guennebaud, G. and Gross, M., 2009. Feature Preserving Point Set Surfaces based on Non-Linear Kernel Regression. Computer Graphics Forum 28(2), pp. 493–501.

Pearson, K., 1901. On lines and planes of closest fit to systems of points in space. Philosophical Magazine 2(6), pp. 559–572.

Schnabel, R., Wahl, R. and Klein, R., 2007. Efficient ransac for point-cloud shape detection. Computer Graphics Forum 26(2), pp. 214–226.

Scholkopf, B. and Smola, A. J., 2001. Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge, MA, USA.

Su, P. and Drysdale, R. L. S., 1995. A comparison of sequential delaunay triangulation algorithms. In: Proceedings of the eleventh annual symposium on Computational geometry, SCG '95, ACM, New York, NY, USA, pp. 61–70.

Tikhonov, A. N., 1943. On the stability of inverse problems. Doklady Akademii Nauk SSSR 39 (5) p. 195198.

Wahba, G., 1990. Spline models for observational data. CBMS-NSF Regional Conference Series in Applied Mathematics, Vol. 59, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA.

Walder, C., Schölkopf, B. and Chapelle, O., 2007. Implicit surfaces with globally regularised and compactly supported basis functions. Advances in Neural Information Processing Systems 19 pp. 273–280.

Wurm, K. M., Hornung, A., Bennewitz, M., Stachniss, C. and Burgard, W., 2010. OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems. In: Proc. of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation, Anchorage, AK, USA.