Supplementary Material – Semantic 3D Reconstruction with Finite Element Bases

Audrey Richard[1,†]
audrey.richard@geod.baug.ethz.ch
Christoph Vogel[2,†]
christoph.vogel@icg.tugraz.at
Maroš Bláha[1]
maros.blaha@geod.baug.ethz.ch
Thomas Pock[2,3]
thomas.pock@icg.tugraz.at
Konrad Schindler[1]
konrad.schindler@geod.baug.ethz.ch

[1] Photogrammetry & Remote Sensing ETH Zurich, Switzerland
[2] Institute of Computer Graphics & Vision TU Graz, Austria
[3] Austrian Institute of Technology

[†] shared first authorship

This document provides supplementary information to support the main paper. It is structured as follows: Sec. 1 gives more information about the data and pre-processing used in our experiments, not mentioned in the paper due to lack of space. We hope that the added details will help readers to better appreciate the experimental results. In Sec. 2 we show complementary results obtained with the proposed Lagrange FEM method on other datasets, as well as the full large-scale reconstruction of the city of Enschede. Sec. 3 contains technical details and formal proofs that had to be omitted in the paper. Finally, Sec. 4 discusses our formalism for the case of the Raviart-Thomas basis (instead of Lagrange P1), leading to piecewise constant labels. We also show results in 2D and 3D and a comparison to those obtained with the Lagrange basis.

1 Input Data

For our real-world experiments, we start from aerial images, *cf*. Fig. 1. To mitigate foreshortening and occlusion, images are acquired in a *Maltese cross* configuration, with four oblique views in addition to the classical nadir view. We orient the images with VisualSFM [10], create depth maps from neighbouring views with Semi-global Matching [3, 8], and predict pixel-wise class-conditional probabilities with a MultiBoost classifier [1]. The classifier is trained on a few hand-labeled images, using the same features as [2]: raw RGB-intensities in a 5×5 window, and 19 geometry features (height, normal direction, anisotropy of structure tensor, *etc*.) derived from the depth map.

2 Additional Visualizations

We have tested our semantic reconstruction method on several (synthetic) 2D and (real) 3D datasets. Here we provide additional examples to give the reader an impression of the

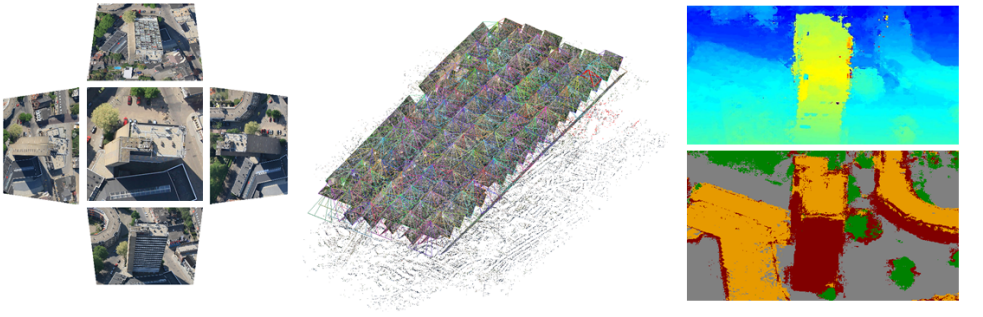

Figure 1: Input data. *Left:* aerial input images for one position (four oblique views to the north, east, south, west, and a nadir image). *Middle:* oriented image block. *Right:* depth map and class probabilities (visualised by maximum-likelihood labels).

variety of cases tested in our evaluation. We apply the same prior models as for our 3D reconstructions. We prefer flat, horizontal structures in the model for the following label transitions: *ground-freespace*, *ground-building*, *building-roof* and *roof-freespace*. The second prior applies to the transition *building-freespace* and prefers vertical boundaries. Fig. 2 shows examples for different degradations of the synthetic input (many more cases were tested). In the top row, we simulate imperfect classifier input by adding noise to the semantic class likelihoods. In our experience, the method is still able to reconstruct the geometry quite well, but sometimes assigns the wrong label. A closer inspection reveals that, locally, the *roof* and *building* classes are confused in locations where the class likelihoods are significantly wrong. The global geometry and labels in other regions remain unaffected. The second row gives an example of missing input data, a frequent situation in the real world, due to occlusions and constraints on camera placement. Fortunately, missing data does not seem to greatly challenge our method. In fact, our method is specifically designed to work well for these cases and complete the outline, relying on the prior assumptions about pairwise class transitions and class-specific local shape. In the last row we utilise only a sparse control mesh, even near the surface. The method can still recover the geometry, but struggles to determine the correct semantic labeling near the (unobserved) *roof-to-building* transition. The adaptive version of our method is designed to avoid exactly that case. It refines the *control mesh* near the predicted transitions, effectively increasing the resolution at the most promising locations.

Fig. 3 shows city models obtained from two additional aerial datasets (Zürich, Switzerland and Dortmund, Germany), and a further patch from Enschede. These results qualitatively illustrate that our method works for different image sets and architectural layouts.

Finally, we show the complete semantic 3D reconstruction of Enschede. Fig. 4 shows the model rendered in an oblique view, together with the corresponding viewpoint in *Google Earth*, to illustrate its accuracy and high level-of-detail.

3 Proofs

In this section we give the technical proofs promised in the main paper, as well as further details about the optimisation. We start with a discussion of the extension to non-metric energies, and its consequences on the equivalence of continuous and discrete models.

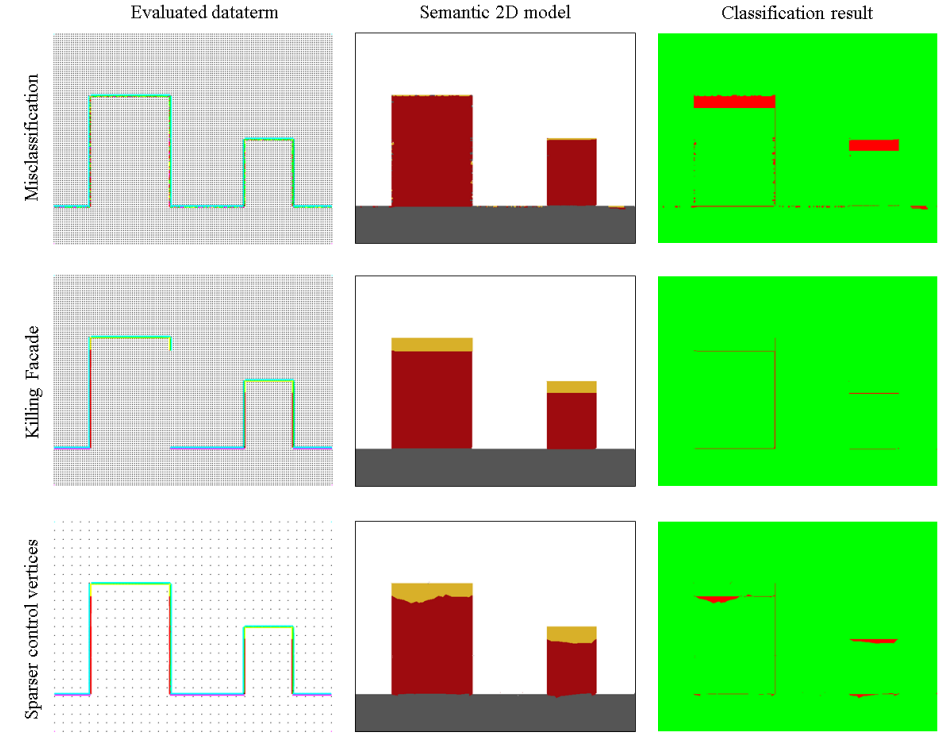

Figure 2: Example scenes of our 2D data set and results obtained with our Lagrange FEM method. *Left:* Data term at vertices of the *control mesh*. Colors for the data cost indicate: *free space/empty space* (cyan), *building* (red), *ground* (pink), *roof* (yellow), *occupied space* (green) and *no data cost* (black). *Middle:* Semantic 2D model. *Right:* Classification result, misclassified pixels are depicted in red.

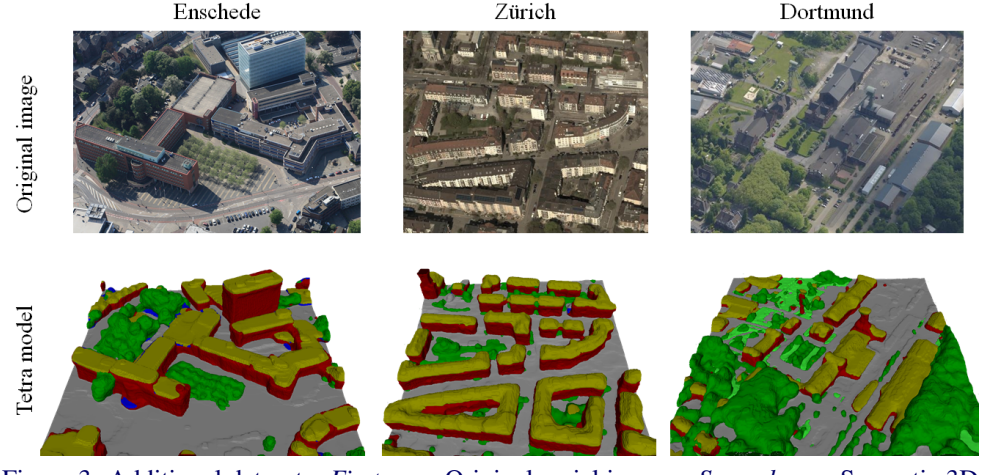

Figure 3: Additional datasets. *First row:* Original aerial images. *Second row:* Semantic 3D models obtained with the Lagrange FEM method for Enschede (left), Zürich (middle) and Dortmund (right). For the latter, light green denotes an additional class *grass and agricultural fields*.

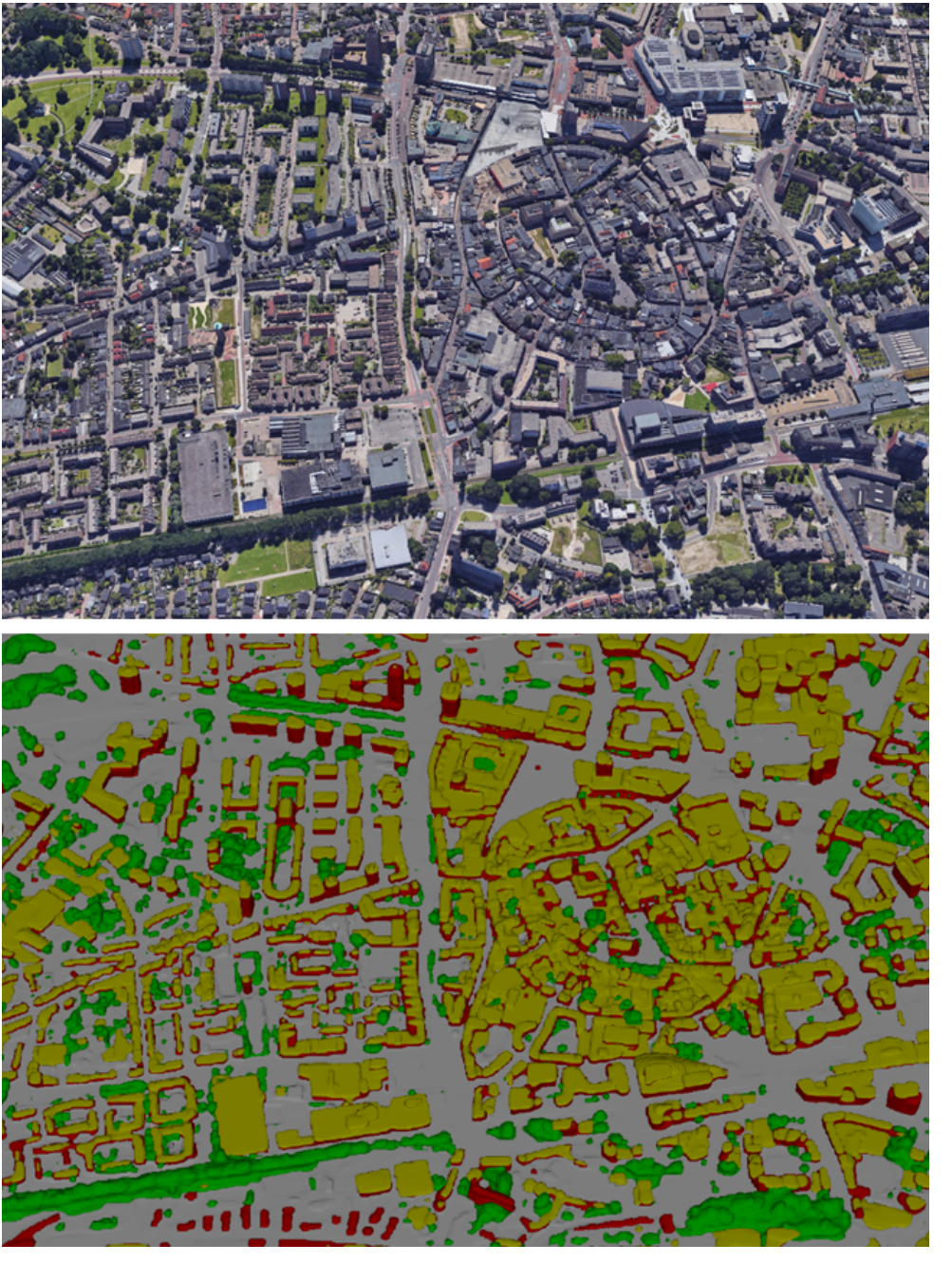

Figure 4: Large-scale semantic 3D reconstruction of Enschede (Netherlands), computed from aerial images with our Lagrange FEM method. *Top:* View from *Google Earth* (not used during reconstruction). *Bottom:* Our model from matching viewpoint.

3.1 Non-metric priors: continuous vs. discrete

The main message of this section is that a non-metric model does not exist in the continuous view, unless one imposes additional constraints on the function spaces. We briefly explain why: Let's look at the transition boundary between two labels i and k. Without additional constraints, one can always introduce a zero set with label j between the two, *i.e.*, a set with Lebesgue measure 0 in the domain space. If the transition costs are not metric, then the cost for the label pair $\{i,k\}$ is potentially higher than the sum of the costs for $\{i,j\}$ and $\{j,k\}$. Inserting the zero set will avoid that extra cost and the energy will be underestimated. In other words, let S be a segmentation of Ω into regions S^i and S^k, labeled with i and k respectively. Assume further that their costs do not fulfill the triangle inequality *w.r.t.* another label j. Then one can find a sequence of segmentations $S^n := \{S^n_i, S^n_j, S^n_k\}$ with $S = \lim_{n\to\infty}(S^n)$: the label j disappears in the limit, such that $\lim_{n\to\infty}\inf E(S^n) < E(S)$. Hence, metric transition costs are a necessary condition for the lower semi-continuity of the energy functional. Methods that try to resolve the issue with additional constraints on the function space, for instance by demanding Lipshitz continuity of the labeling functions, are an active research area, *e.g.* [4], but are beyond the scope of this work.

The above conceptual problem does have consequences for a practical implementation: Any discretisation of the domain will ultimately consist only of a finite number of elements of measurable (>0) volume. Thus, the label j in the example will not disappear completely from the solution, and the computed energy matches the solution. In practice, one can simply prescribe a minimum edge length in the tesselation, since one cannot refine infinitely. Note that this also constrains the Lipshitz constant of the labeling functions; they are restricted to values between 0 and 1, such that the Lipshitz constant of functions $f \in P^1(M)$ defined on the mesh $M = \{V,F,S\}$ is bounded by $\min_{v\in s, s\in S}||J_v||$, *cf.* (1). Because we utilise a Delaunay triangulation/tetrahedralisation of the domain and also limit the minimal dihedral angle, a further constraint on the edge length implies a bound on the Lipshitz constant. Note also, our analysis implies that a discrete solution in the non-metric setting does not have a continuous counterpart, and consequently investigations of the limiting case, *i.e.*, convergence analysis after infinite refinement of the tesselation, are futile.

3.2 Gradient in the Lagrange basis

We show that gradients of functions in the P1 (Lagrange) basis are constant per simplex s and given by:

$$\nabla\phi_s = \sum_{v\in s}\phi_v J_v \tag{1}$$

Here, the coefficients $\phi_v \in \mathbb{R}$ and $J_v \in \mathbb{R}^d$ denote a vector of length $\frac{|f_v|}{|s|d}$, normal to the face f_v opposite to vertex v, and pointing inwards towards the center of the simplex. Recall that $|f_v|$ is the area of face f_v and $|s|$ is the volume of simplex s.

The gradient can be obtained with basic algebra. First, notice that the gradient of ϕ_s in (1) has to fulfill $\langle v_l - v_k, \nabla\phi_s\rangle = \phi_{v_l} - \phi_{v_k}$, meaning that integration along the edge leads to the respective change in ϕ_s. After collecting a sufficient number of linear equations of this form, one can directly solve the resulting linear system. Since J_v is, by definition, orthogonal to all edges that do not involve vertex v, we arrive at (1).

Formally, we pick one vertex v of simplex s and compile for $l = 1\ldots d$ ($v_l \neq v$) equations of the form $\langle v_l - v, \nabla\phi_s\rangle = \phi_{v_l} - \phi_v$. By construction, $\langle J_{v_l}, v_k - v\rangle = \delta_{k=l}$. The vector J_{v_l} is

normal to face f_{v_l}. The scalar product of the edge $(v_l - v)$ and the normal is the "height" within the simplex, so with the chosen scaling of J_{v_l} we have $\langle J_{v_l}, v_l - v_k \rangle = 1$ for any $k \neq l$.

Thus multiplying each side of our equation system by a matrix with the vectors $J_{v_l}, l = 1 \dots d$ as columns leads to: $\nabla \phi_s = \sum_l J_{v_l}(\phi_{v_l} - \phi_v)$. If we can show that $\sum_l J_{v_l} = -J_v$, then we arrive at the desired expression (1). For $v_k, v_j \neq v$, $\langle \sum_l J_{v_l}, v_j - v_k \rangle = \langle J_{v_l}, v_j \rangle - \langle J_{v_k}, v_k \rangle = 0$ and $\langle \sum_l J_{v_l}, v_k - v \rangle = \langle J_{v_k}, v_k - v \rangle = 1$. All equations are also fulfilled by J_v in place of $\sum_l J_{v_l}$, which concludes the proof.

3.3 Data Term for Lagrange basis

We again start from the ideas in the main paper. We have to convert the continuous data costs ρ^i into discrete form (in a practical implementation, "continuous" means that the cost can be evaluated at any $z \in \Omega$). In our basis representation, we can get discrete cost values for the basis elements by convolving the continuous cost with the respective basis function. For simplicity, we consider the P1 basis function here. Thus, we seek a cost per vertex ρ^i_v. In detail we obtain:

$$\int_\Omega x^i(z)\rho^i(z)\mathrm{d}z = \sum_s \int_s x^i_s(z)\rho^i(z)\mathrm{d}z = \sum_s \int_s \sum_{v \in s} \phi_v p^1_{s,v}(z)\rho^i(z)\mathrm{d}z = \sum_{v \in V} \phi_v \underbrace{\left(\sum_{s \in N(v)} \int_s p^1_{s,v}(z)\rho^i(z)\mathrm{d}z \right)}_{:=\rho^i_v} = \langle \rho^i_v, x^i_v \rangle. \tag{2}$$

To numerically compute ρ^i_v, we sample ρ^i at a finite number of locations $z \in \Omega$. For each z we determine into which simplex s it falls, and accumulate the contributions of $\rho^i(z)$ over all $i = 1 \dots m$, weighted by their barycentric coordinates. The final step is to scale ρ^i_v by $\sum_{s \in N(v)} |s|/d$ and divide by the sum of weights assigned to vertex v. In other words, we compute the sample mean and scale it by the area covered by the vertex. In our current implementation ρ is sampled at regular grid points, without importance sampling. This simple strategy is indeed very similar to the method employed in [2, 7]. There, the data cost is evaluated on a regular grid, by reprojecting grid vertices into each image, computing the data term, and adding its respective contribution to the grid location. Such a "per-voxel accumulation" is equivalent to integrating the data cost within the respective Voronoi-area of a vertex in the dual grid: the latter is proportional to the number of regular samples that fall into a Voronoi-cell and therefore have the respective vertex as nearest neighbour. Hence, summing the individual contributions directly corresponds to integrating the data term within the Voronoi region.

3.4 Grid vs. P1

Here, we detail why the grid-based version with finite differences (corresponding to [7]) can be seen as an approximation of our proposed FEM discretisation with P1 basis elements, if the vertices (cells) are aligned in a regular grid. Without loss of generality we consider a grid of edge length 1, and note that in this case the gradient for a function $f : \Omega \subset \mathbb{R}^d \to \mathbb{R}$ at a grid point x, evaluated with forward differences becomes:

$$\nabla f = (f_{x+e_1} - f_x, \dots, f_{x+e_d} - f_x)^\mathsf{T} = \sum_i e_i f_{x+e_i} - \sum_i e_i f_x = \sum f_{x+e_i} J_{x+e_i} + f_x J_x, \tag{3}$$

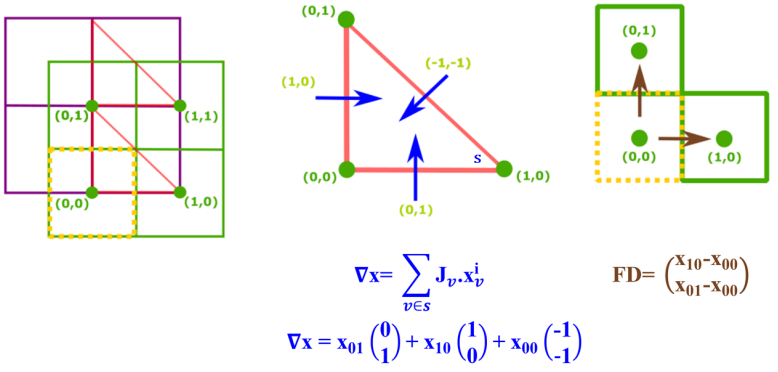

Figure 5: *Left:* Grid (*green*) and simplex mesh (*red*) cover the same domain, but are offset against each other. Grid centers correspond to vertex positions. The gradient in a triangle (*middle*) corresponds to the gradient computed with forward differences (*right*).

with e_i the unit vector in direction i. We have used the identity $e_i = J_{x+e_i}$, according to the definition in Sec. 3.2, and obtain the last equality from $\sum_l J_{v_l} = -J_v$, *cf*. Sec. 3.2. This is exactly the formula for the gradient of the corresponding P1 function in the simplex defined by the vertices $\{x, x+e_i\}_{i=1}^d$. Accordingly, if implemented as finite differences, the constraints on the dual vector field λ, see Eq. (2) from the paper, are only checked within the respective simplex, but not in the whole domain (1/2 of the domain in 2D; 1/6 in 3D). Note also that, with grid-aligned vertices, the simplex in question cannot be part of a partition of $\Omega \subset \mathbb{R}^d$, unless $d \leq 2$: edges of adjacent faces would intersect.

Fig. 5 illustrates the specific case with $d = 2$. On the left, the regular grid (*green*) and the triangle (simplex, *red*). Grid centers correspond to vertices in the (triangle-)mesh. The grid corresponds to the discretisation used in [7], whereas the simplex mesh is used in this paper. The gradient of the lower left triangle for the simplex mesh corresponds exactly to the one computed via forward differences as shown in (3). Consequently, discretisation via finite differences is a special case of our method, where the elements are layed out on a regular grid, and the constraints are tested only in the upper right triangle (in the 2D case).

3.5 Adaptiveness

We have stated in the paper that our formulation is adaptive, in the sense that a hierarchical refinement of the tesselation can only decrease the energy. We have also explained a way to find the refined tesselation of Ω, by introducing additional vertices and splitting simplices s, such that no faces $f \in F$ are flipped; and we have put forward a procedure to initialise the new variables. Here, we formally prove that the described scheme is sound.

Let $(x^*, \lambda^*, \theta^*)$ be the solution for a triplet $M = \{F, V, S\}$. And let $\hat{M} = \{\hat{F}, \hat{V}, \hat{S}\}$ be the refined mesh with $V \subset \hat{V}$ and $\forall \hat{s} \in \hat{S}, \exists s \in S$ with $\hat{s} \cap s = \hat{s}$. Furthermore, we define the sets $\bar{V} = \hat{V} \setminus V$ and $\bar{S} = \hat{S} \setminus S$ to denote newly introduced vertices and simplices. Our construction works by induction, *i.e.* we introduce one vertex v at a time. The vertex is assumed to lie in simplex $s \in S$, $s = \{v_k\}_{k=1}^{d+1}$, which is split into simplices $\{\bar{s}_k\}_{k=1}^{d+1}$ with $\bar{s}_k \cap \bar{s}_l = \emptyset, \forall k \neq l$. By definition, vertex $\bar{v}$ has the barycentric coordinates $\frac{|\bar{s}_k|}{|s|}$, *i.e.* $\bar{v} = \sum_{k=1}^{d+1} \frac{|\bar{s}_k|}{|s|} v_k$.

We initialize the labeling variables at new vertices $\bar{v} \in \bar{V}$ via barycentric interpolation: $x^i_{\bar{v}} = \sum_{v_k \in s} \frac{|\bar{s}_k|}{|s|} x^i_{v_k}$. Dual variables of the new simplices $\bar{s}_k$, and also the transition variables

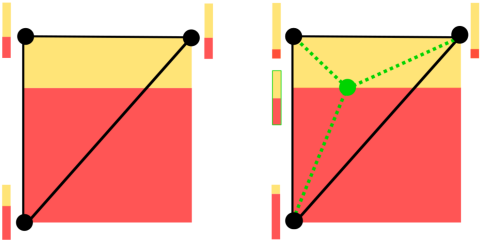

Figure 6: Updated data term after adding a new vertex.

x^{ij}, are simply copied from their enclosing simplex s: $\lambda^i_{\bar{s}_k} := \lambda^i_s$, $\theta^i_{\bar{s}_k} := \theta^i_s$ and $x^{ij}_{\bar{s}_k} := x^{ij}_s$. The data terms at the new vertex $\bar{v}$, as well as at the other vertices v_k of simplex s, are (re)computed following (2), see also Fig. 6. We call the new variables $\bar{x}, \bar{\lambda}, \bar{\theta}$ and claim that $E_M(x^*, \lambda^*, \theta^*) = E_{\hat{M}}(\bar{x}, \bar{\lambda}, \bar{\theta})$ and that $\bar{x}, \bar{\lambda}, \bar{\theta}$ is feasible. The latter is trivially the case (*cf*. Eq. (11) from the paper); transition variables remain positive by construction, the newly introduced labeling variables still fulfill the simplex constraints, and they induce the same gradients in the new simplices as in the parent simplex. The same applies for the cost of the regulariser in the new simplices: $\sum_{i<j} |s| ||x^{ij}_s - x^{ji}_s||_{W^{ij}} = \sum_{i<j} \sum_{\bar{s}_k \in s} |\bar{s}_k| ||x^{ij}_{\bar{s}_k} - x^{ji}_{\bar{s}_k}||_{W^{ij}}$. More interesting is the data cost. We introduce the notation $\rho_{v,s} := \int_s p^1_{s,v}(z) \rho^i(z) \mathrm{d}z$ for the data cost at vertex v, originating from the integral over simplex s. By induction, we need to only verify the following equality for simplex s, which is split into $\{\bar{s}_k\}_{k=1}^{d+1}$:

$$\sum_k x^i_{v_k} \rho_{v_k,s} = \sum_k x^i_{v_k} \sum_{j \neq k} \rho_{v_k,\bar{s}_j} + \sum_j x^i_{\bar{v}} \rho_{\bar{v},\bar{s}_j} = \sum_k x^i_{v_k} \sum_{j \neq k} \rho_{v_k,\bar{s}_j} + \frac{|\bar{s}_k|}{|s|} \sum_j x^i_{v_k} \rho_{v,\bar{s}_j} \tag{4}$$

Recall we have "old" vertices v_k, $s = \{v_k\}_{k=1}^{d+1}$ and a new vertex $\bar{v}$. According to (4), we must verify:

$$\rho_{v_k,s} = \sum_{j \neq k} \rho_{v_k,\bar{s}_j} + \frac{|\bar{s}_k|}{|s|} \sum_j \rho_{v,\bar{s}_j} \Leftrightarrow \int_s \rho(z) p^1_{s,v_k}(z) \mathrm{d}z = \int_s \rho(z) \sum_{j \neq k} p^1_{\bar{s}_j,v_k}(z) + \frac{|\bar{s}_k|}{|s|} \sum_j p^1_{\bar{s}_j,v}(z) \mathrm{d}z. \tag{5}$$

It is sufficient to show

$$p^1_{s,v_k}(z) = \sum_{j \neq k} p^1_{\bar{s}_j,v_k}(z) + \frac{|\bar{s}_k|}{|s|} \sum_j p^1_{\bar{s}_j,v}(z) \quad \forall z \in s. \tag{6}$$

The right hand side represents a linear function for each $\{\bar{s}_k\}_{k=1}^{d+1}$. We can check if both sides agree on $d+1$ points in each simplex, which is easy to verify. The locations we check – substitute z on both sides of Eq. (6) – are $\{v_k\}_{k=1}^{d+1}$ and v. These are the defining vertices of the $d+1$ simplices $\{\bar{s}_k\}_{k=1}^{d+1}$. Left and right hand side vanish, except for v and v_k. Finally, we get $\frac{|\bar{s}_k|}{|s|}$ for v and 1 for v_k on both sides.

In our adaptive version, we directly follow the proof and split simplices with the introduction of a single new vertex. We emphasise again that this splitting schedule is merely a proof of concept. The FEM discretisation allows for more sophisticated refinement schemes, *e.g*., along the lines of [6], or flipping edges according to the energy functional, *etc*.

3.6 Optimisation

The energy (11) from the paper is given in primal-dual form, optimisation with existing tools is straight-forward. We apply the minimisation scheme of [5], with pre-conditioning

[2]. Internally, that algorithm however requires the projection onto the Wulff-shapes W^{ij}, which is slightly more involved.

3.6.1 Proxmap for the Minkowski sum of convex sets

Recall that, per label pair $\{i,j\}$, our Wulff-shapes are of the form $W^{ij} := \Psi^{ij} \oplus B^2_{\kappa^{ij}}$. They are the Minkowski sum of two simple convex sets. Recall that the Ψ^{ij} encode the direction dependent likelihood of a certain label transition. In our case, all Wulff-shapes permit a closed form projection scheme, such that we solve the following sub-problem as proximal step, independently per simplex s:

$$\arg\min_{x^{ij},x^{ji}} \frac{1}{2}||x^{ij}-\overline{x^{ij}}||^2+\frac{1}{2}||x^{ji}-\overline{x^{ji}}||^2+\sup_{w\in\Psi^{ij}\oplus B^2_{\kappa^{ij}}} w^\mathsf{T}(x^{ij}-x^{ji})+\iota_{\geq 0}(x^{ij})+\iota_{\geq 0}(x^{ji}). \tag{7}$$

For the following derivation we rename the two sets $W_1 := \Psi^{ij}$ and $W_2 := B^2_{\kappa^{ij}}$. In order to decouple the argument within the regulariser, we introduce auxiliary variables $\{y_k,z_k\}_{k=0}^2$ and additional Lagrange multipliers $\{\mu_k,\lambda_k\}_{k=0}^2$, and replace x^{ij} and x^{ji} respectively:

$$\begin{aligned}\min_{x^{ij},x^{ji},y_k,z_k}\ \max_{\mu_k,\lambda_k}\ &\frac{1}{2}||x^{ij}-\overline{x^{ij}}||^2+\frac{1}{2}||x^{ji}-\overline{x^{ji}}||^2+\\ &\sum_{k\in\{1,2\}}\sup_{w\in W_k} w^\mathsf{T}(y_k-z_k)+\iota_{\geq 0}(y_0)+\iota_{\geq 0}(z_0)-\sum_{k=0}^2\lambda_k^\mathsf{T}(x^{ij}-y_k)-\mu_k^\mathsf{T}(x^{ji}-z_k).\end{aligned} \tag{8}$$

Optimality w.r.t. x^{ij},x^{ji} implies:

$$x^{ij}=\overline{x^{ij}}+\sum_{k=0}^2\lambda_k \ \text{ and } \ x^{ji}=\overline{x^{ji}}+\sum_{k=0}^2\mu_k, \tag{9}$$

which, after reinserting into (8), leads to:

$$\begin{aligned}\min_{y_k,z_k}\max_{\mu_k,\lambda_k}\ &\frac{-1}{2}||\sum_{k=0}^2\lambda_k-\overline{x^{ij}}||^2+\frac{-1}{2}||\sum_{k=0}^2\mu_k-\overline{x^{ji}}||^2+\\ &\sup_{w_1\in W_1,w_2\in W_2} w_1^\mathsf{T}(y_1-z_1)+w_2^\mathsf{T}(y_2-z_2)+\iota_{\geq 0}(y_0)+\iota_{\geq 0}(z_0)+\sum_{k=0}^2\lambda_k^\mathsf{T}y_k+\mu_k^\mathsf{T}z_k\ .\end{aligned} \tag{10}$$

Applying Fenchel-duality yields:

$$\begin{aligned}\max_{\mu_k,\lambda_k}\min_{z_k}\ &\frac{-1}{2}||\sum_{k=0}^2\lambda_k-\overline{x^{ij}}||^2+\frac{-1}{2}||\sum_{k=0}^2\mu_k-\overline{x^{ji}}||^2\\ &-\iota_{W_1}(-\lambda_1)-\iota_{W_2}(-\lambda_2)-\iota_{\leq 0}(-\lambda_0)-\iota_{\leq 0}(-\mu_0)+\sum_{k=1}^2(\lambda_k+\mu_k)^\mathsf{T}z_k.\end{aligned} \tag{11}$$

The latter summand requires $\lambda_1=-\mu_1$ and $\lambda_2=-\mu_2$:

$$\min_{\mu_0,\lambda_k}\frac{1}{2}||\sum_{k=0}^2\lambda_k-\overline{x^{ij}}||^2+\frac{1}{2}||\sum_{k=1}^2\lambda_k+\overline{x^{ji}}-\mu_0||^2+\iota_{W_1}(-\lambda_1)+\iota_{W_2}(-\lambda_2)+\iota_{\geq 0}(\lambda_0)+\iota_{\geq 0}(\mu_0). \tag{12}$$

In this last form, we can apply a few iterations of block coordinate descent on the dual variables and recover the update for x^{ij},x^{ji} from (9).

4 Raviart-Thomas basis

4.1 Methodology

In this section, we show how to discretise the convex relaxation, Eq. (2) from the paper, for the case of the Raviart-Thomas basis. For convenience, we restate the energy:

$$\min_{x^i}\max_{\lambda^i}\sum_i\int_\Omega \rho^i(z)x^i(z)+\langle x^i(z),\nabla\cdot\lambda^i(z)\rangle \mathrm{d}z,\ \text{s.t. } \lambda^i(z)-\lambda^j(z)\in W^{ij},\sum_{i=1}^m x^i(z)=1, x^i(z)\geq 0. \quad (13)$$

The Raviart-Thomas basis is chosen as a strong contrast to the (preferred) Lagrange basis. With Raviart-Thomas functions, we model the dual functions λ in (13), within our trial space. The Raviart-Thomas $\mathrm{RT}^k(M)$ basis functions describe a *div*-conforming polynomial basis of order $k+1$, *i.e.* the divergence of the modeled vector field is continuous across simplices. We again discretise on a simplex mesh $M=\{F,V,S\}$ with vertices $v\in V, v\in\mathbb{R}^d$; faces $f\in F$ defined by d vertices; and simplices $s\in S$ defined by $d+1$ vertices, which partition Ω: $\cup_k s_k=\Omega, s_l\cap s_k=f_{k,l}\in F$.

$$RT^0(M):=\{p:\Omega\to\mathbb{R}^d|\phi(x):=\sum_{s\in S}\phi_s(x) \text{ with } \phi_s(x):=c_s x+d_s, c_s\in\mathbb{R}, d_s\in\mathbb{R}^d,$$
$$\text{if } x\in s \text{ and } 0 \text{ else, and } \phi_s(x) \text{ is continuous for } x\in f_v(s) \text{ in direction } \nu^s_{f_v}\}. \quad (14)$$

Here, we have used $\nu^s_{f_v}$ to denote the (outward-pointing) normal of face f_v of simplex s. By convention the face f_v is located opposite the vertex v. We construct our linear basis with functions that are defined for each face f_v in a simplex s, and can be described in a *local* form as:

$$\phi^0_{s,v}(x):=(x-v)\frac{|f_v|}{|s|d} \quad \text{if } x\in s \text{ and } 0 \text{ else},$$

where we again let $|f_v|$ denote the area of the face and $|s|$ the volume of the simplex. Let $\nu^s_{f_v}$ be the normal of face f_v in simplex s, then the basis functions fulfill:

$$\langle\phi^0_{s,u}(x),\nu^s_{f_v}\rangle:=[u=v]\ \forall x\in f_v, \quad (15)$$

with $[\cdot]$ denoting the Iverson bracket.

These basis functions make up the *global* function space by enforcing a consistent orientation. For each face f we can distinguish its two adjacent simplices s^+ and s^-, by analysing the scalar product of the vector $\mathbf{1}$ and the normal $\nu^{s\pm}_f$ of the shared face f (by convention again pointing outwards of the respective simplex). W.l.o.g., we define $\mathrm{si}^{s^\pm}_f:=\mathrm{sign}\langle\nu^{s^\pm}_f;\mathbf{1}\rangle$, *i.e.* $s^+_f\nu^{s^+}_f=s^-_f\nu^{s^-}_f$. The global basis functions per face f_v are then given by:

$$\phi^0_{s,v}:=\begin{cases}\mathrm{si}^s_{f_v}(x-v)\frac{|f_v|}{|s|d} & \text{if } x\in s\\ 0 & \text{else.}\end{cases} \quad (16)$$

In each simplex, our vector-field $\phi_s(x)\in\mathbb{R}^d$ can then be defined in the following manner, with coefficients $\phi_{f_v}\in\mathbb{R}$:

$$\phi_s(x):=\sum_{v\in s}\phi_{f_v}\phi^0_{s,v}(x).$$

By construction, *cf*. (15),(16), the vector-field is continuous along a face $x \in f$ in direction of the face normal ν_f (of arbitrary, but fixed orientation), *i.e*. for neighbouring faces s^+ and s^- we have:

$$\langle \phi_{s^\pm}(x), \nu_f \rangle = \mathrm{si}_f^{s^\pm} \langle x - v^\pm, \nu_f \rangle \frac{|f|}{|s^\pm| d} \phi_{f_{v^\pm}} = \phi_{f_{v^\pm}}. \tag{17}$$

Here, v^+ is the vertex in simplex s^+ opposite to the shared face, and v^- is the vertex in simplex s^-. Thus, our function in ϕ is a RT function *iff* for all neighbouring faces $s^\pm$ we have $\phi_{f_{v^+}} = \phi_{f_{v^-}}$. In other words, basis coefficients only exist for faces of the simplices.

The variables we are interested in are the labeling functions x^i, which are members of our test function space, composed of piecewise constant functions per simplex:

$$U^0(M) := \{u : \Omega \to \mathbb{R} | u(x) := \sum_{s \in S} u_s(x), \text{ with } u_s(x) = u_s \text{ if } x \in s \text{ and } 0 \text{ else}\}. \tag{18}$$

Before we can utilise our new basis to discretise (13) we need a way to enforce the constraints on our dual variables $\lambda^i(z) - \lambda^j(z) \in W^{ij}$ for all $z \in \Omega$. It is sufficient to enforce the constraints on the dual functions λ in (13) only at the face midpoints $z_{f_v} := 1/d \sum_{w \in f_v} w$ of faces $f_v \in s$. This ensures the constraints are also valid for any point in the simplex s. Because the Wulff shapes are convex, it is sufficient to prove that a vector field $\phi(x) \in RT^0(M), \phi(x) := \sum_{s \in S} \phi_s(x)$ at any point $x \in s$ can be written as a convex combination of the values at the face midpoints:

$$\phi_s(x) = \sum_{f_v \in s} \alpha_{z_{f_v}} \phi(z_{f_v}), \text{with} \sum_{f_v \in s} \alpha_{z_{f_v}} = 1.$$

After some elementary algebra it turns out that, if $x = \alpha_i v_i$, then $\alpha_{z_{f_v}} := (1 - d \cdot \alpha_i)$ encode this convex combination. Furthermore, the value of ϕ_s at a location $x \in s$ can be found by linear combination of basis coefficients at the vertices of s:

$$\phi_s(x) = \sum_{v \in s} \mathrm{si}_{f_v}^s (x - v) \frac{|f_v|}{|s| d} \phi_v. \tag{19}$$

4.2 Discretisation

With these relations, we can discretise the energy (13) for labeling functions $x^i \in U^0(M)$ and dual vector-field $\lambda^i \in RT^0(M)$. First, we convert the continuous data costs ρ^i into a cost per simplex ρ_s^i, which can again be achieved by convolving the cost with the respective (per simplex constant) basis function: $\rho_s^i := \int_s u_s(z) \rho^i(z) \mathrm{d}z = \int_s \rho^i(z) \mathrm{d}z$. In practice, the integral is computed via sampling. Next, we discretise the second part of our energy with the help of the divergence theorem and (17):

$$\int_\Omega x^i(z) \nabla \cdot \lambda^i(z) \mathrm{d}x = \sum_{s \in S} \int_S x_s^i \nabla \cdot \lambda(z) \mathrm{d}x = \sum_{s \in S} \int_{\partial S} x_s^i \langle \lambda(z), \nu(z) \rangle \mathrm{d}z = \sum_{v \in s, s \in S} x_s^i \lambda_{f_v}^i |f_v| \mathrm{si}_{f_v}^s \tag{20}$$

As shown, we need to verify the constraints only at face midpoints z_{f_v}. The vectors $\lambda_s(z_{f_v})$ are linear in the basis coefficients for any $z \in \Omega$, and the discretised version of (13) becomes

$$\min_{x^i} \max_{\lambda^i} \sum_{s \in S} \rho_s^i x_s^i + \sum_{v \in s, s \in S} x_s^i \lambda_{f_v}^i |f_v| \mathrm{si}_{f_v}^s, \text{ s.t. } \lambda_s^i(z_{f_v}) - \lambda_s^j(z_{f_v}) \in W^{ij}, \; x_s^i \in \Delta \; \forall i < j, v \in s, s \in S. \tag{21}$$

Here, we let Δ encode the unit simplex. Finally, for every simplex s, we replace the constraint set $\sum_{i<j} \lambda_s^i(z_{f_v}) - \lambda_s^j(z_{f_v}) \in W^{ij}$ in the same manner as for the Lagrangian basis. We introduce auxiliary variables and Lagrange multipliers $y^{ij}_{s,f_v}, \forall i < j$, and exploit Fenchel-Duality to obtain

$$\begin{aligned}
&\max_{\lambda_s^i} \min_{y_s^{ij}} \sum_{v\in s}\sum_{i<j} ||y^{ij}_{s,f_v}||_{W^{ij}} - \sum_{v\in s}\sum_i \langle \lambda_s^i(z_{f_v}),\ \sum_{j:i<j} y^{ij}_{s,f_v} - \sum_{j:j<i} y^{ji}_{s,f_v} \rangle = \\
&\max_{\lambda_s^i} \min_{y_s^{ij}} \sum_{v\in s}\sum_{i<j} ||y^{ij}_{s,f_v}||_{W^{ij}} - \sum_{v\in s}\sum_i \lambda^i_{f_v} \left(\frac{|f_v|\mathrm{si}^s_{f_v}}{|s|d} \left[\sum_{\bar{f}\in s} (z_{\bar{f}} - v)^\mathsf{T} \left(\sum_{j:i<j} y^{ij}_{s,\bar{f}} - \sum_{j:j<i} y^{ji}_{s,\bar{f}} \right) \right] \right).
\end{aligned} \tag{22}$$

Furthermore, recall that we use Neumann conditions at the boundary of Ω, which translates into coeffients $\lambda^i_f = 0$, $\forall f \in \partial\Omega$. Combining (21) and (22), we get the (metric) energy for the Raviart-Thomas discretisation:

$$\begin{aligned}
&\min_{x^i, y^{ij}} \max_{\lambda^i} \sum_{s\in S}\sum_i \rho^i_s x^i_s + ||y^{ij}_{s,f_v}||_{W^{ij}} + \iota_\Delta(x^i_s) \\
&\quad + \sum_{v\in s}\sum_i \lambda^i_{f_v} |f_v| \mathrm{si}^s_{f_v} \left(x^i_s - \frac{1}{|s|d} \left[\sum_{\bar{f}\in s} (z_{\bar{f}} - v)^\mathsf{T} \left(\sum_{j:i<j} y^{ij}_{s,\bar{f}} - \sum_{j:j<i} y^{ji}_{s,\bar{f}} \right) \right] \right)
\end{aligned} \tag{23}$$

To extend it to non-metric pairwise costs, as in the Lagrangian case, we need to impose additional assumptions. One possibility is to utilise basis functions for the dual variables, which are continuous in all directions at the faces. In that case, it is only necessary to check the constraints at the faces and not for each face in each simplex, *i.e.* the variables for $y^{ij}_{s^+,f}$ and $y^{ij}_{s^-,f}$ merge into one set. Another possibility is to only force the normal component along the faces of λ to be contained in the Wulff-shapes. In this direction, RT is already continuous and the Lagrange multipliers y can be merged. This line of attack leads to a scheme that is remarkably similar to belief propagation on a Markov random field, in the sense that the discretisation lacks a continuous counterpart to begin with, and may lead to stronger grid artifacts. We stop at this point and leave an investigation of such models to future work.

4.3 2D results

Fig. 7 illustrates the result we obtain with the Raviart-Thomas FEM method (RT). We use the same (perfect) baseline setting as for the Lagrange FEM method (P1) in the main paper. In that setting, the RT method achieves 97.5% of *overall accuracy* and 92.8% of *average accuracy*. While these results confirm that also the RT method is sound, they also show its limitations compared to the Lagrange basis. Simplices not aligned with object boundaries, straddling multiple labels, will necessarily introduce errors in the reconstruction. Note that we do not used edge information to guide the meshing; especially since such information is not available for our target application, semantic 3D reconstruction. We refer to the 3D qualitative comparison (*cf*. Sec. 4.4) for a more detailed analysis of the differences between the two methods.

NB: Further to this manuscript, the supplementary material contains a short **video**, which shows the diffusion of the indicator function over 1000 iterations for both proposed methods.

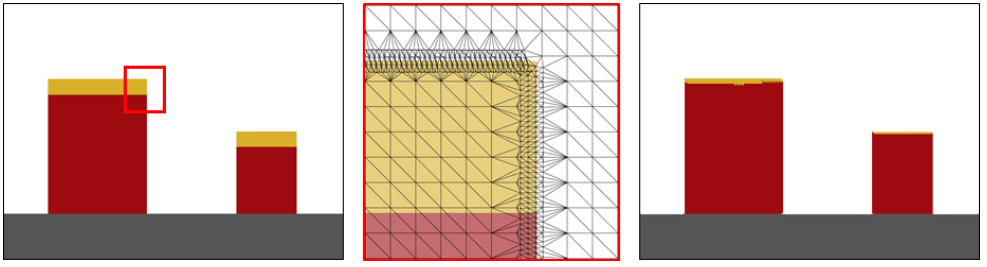

Figure 7: *Left:* Synthetic 2D scene. Colors indicate ground (*gray*), building (*red*) and roof (*yellow*). *Middle:* Zoom of the *control mesh*. *Right:* Reconstructed semantic 2D model.

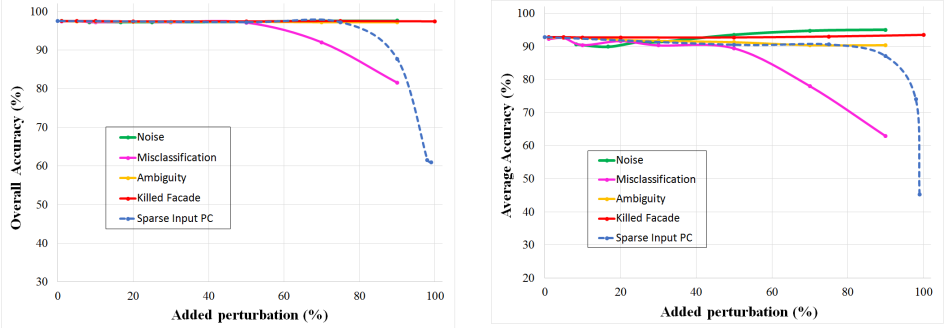

Figure 8: Quantitative evaluation of Raviart-Thomas FEM method *w.r.t.* different degradations of the input data.

We perform also the same series of experiments where we incrementally add different types of perturbations, *cf*. Sec. 5 of the main paper. Fig. 8 shows the corresponding behaviour of our RT method. Generally speaking, both models shows a similar sensitivity to defective inputs, but with a small edge for the Lagrange method, which consistently reaches higher overall accuracy.

4.4 3D results

Fig. 9 shows a quantitative evaluation of the Raviart-Thomas basis, equal to the one of the Lagrange basis presented in the main paper. As before, the colors encode *building* (*red*), *ground* (*gray*), *vegetation* (*green*), *roof* (*yellow*) and *clutter* (*blue*). We summarise the outcomes in Tab. 1. The differences between the Lagrange basis and octree are vanishingly

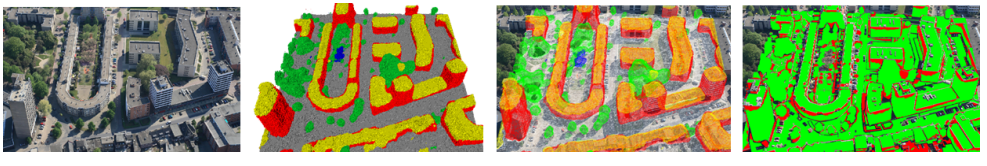

Figure 9: Quantitative evaluation of Scene 1 from Enschede. *Left:* One of the input images. *Middle left:* Semantic 3D model obtained with our Raviart-Thomas FEM method. *Middle Right:* Back-projected labels overlayed on the image. *Right:* Error map, misclassified pixels are marked in red.

small, on the other hand we notice a bigger gap between the Raviart-Thomas basis and octree. We also present a qualitative comparison of the two bases in Fig. 10. The differences are immediately apparent, which confirms the numbers given in Tab. 1. Raviart-Thomas labels entire simplices, so the reconstruction consists of piecewise constant elements. On the contrary, the Lagrangian basis has the advantage that the labeling functions are linear and can be interpreted as (signed) distance functions, such that a smooth iso-surface can be extracted, here done with marching tetrahedra. Despite the piecewise constant reconstruction, the RT basis measures metric quantities – in contrast to, for instance, Markov random fields, where pairwise distances between the simplices would have to be designed explicitly to achieve similar effects.

Data set	Error measure	Tetra P1	Tetra RT	Octree	MB
Scene 1	Overall acc. [%]	84.0	81.9	83.9	82.5
	Average acc. [%]	81.1	79.1	80.6	81.4

Table 1: Quantitative comparison of our two proposed FEM methods with octree model [2] and MultiBoost input data [1].

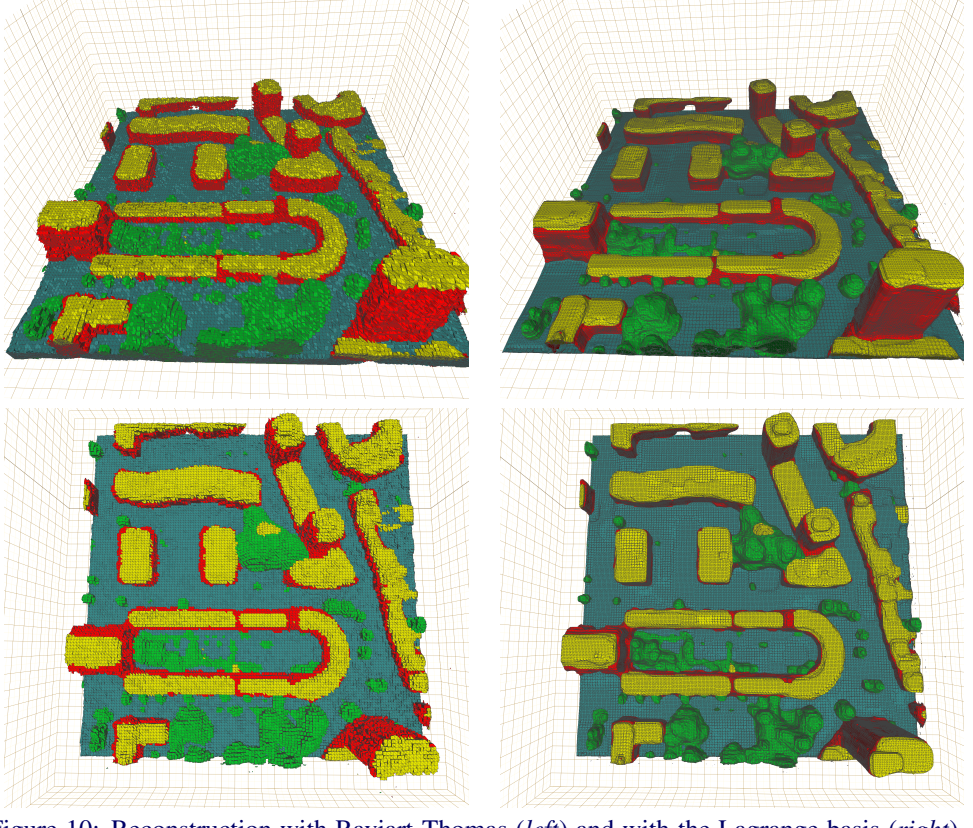

Figure 10: Reconstruction with Raviart-Thomas (*left*) and with the Lagrange basis (*right*). We deliberately select a low resolution, choose flat shading and plot mesh edges, to accentuate the differences. Please refer to the text for details.

References

[1] D. Benbouzid, R. Busa-Fekete, N. Casagrande, F-D. Collin, and B. Kégl. MULTIBOOST: a multi-purpose boosting package. *JMLR*, 2012.

[2] Maros Blaha, Christoph Vogel, Audrey Richard, Jan Dirk Wegner, Thomas Pock, and Konrad Schindler. Large-scale semantic 3d reconstruction: An adaptive multi-resolution model for multi-class volumetric labeling. In *CVPR*, 2016.

[3] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[4] Elie Bretin and Simon Masnou. A new phase field model for inhomogeneous minimal partitions, and applications to droplets dynamics, 2017.

[5] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *JMIV*, 40(1), 2011.

[6] Eitan Grinspun, Petr Krysl, and Peter Schröder. CHARMS: A Simple Framework for Adaptive Simulation. *SIGGRAPH*, 2002.

[7] Christian Häne, Christopher Zach, Andrea Cohen, Roland Angst, and Marc Pollefeys. Joint 3d scene reconstruction and class segmentation. In *CVPR*, 2013.

[8] Heiko Hirschmüller. Stereo processing by semiglobal matching and mutual information. *PAMI*, 2008.

[9] Thomas Pock and Antonin Chambolle. Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In *ICCV*, 2011.

[10] Changchang Wu. VisualSFM: A visual structure from motion system, 2011.