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

Abstract

We propose a novel framework for the discretisation of multi-label problems on arbitrary, continuous domains. Our work bridges the gap between general FEM discretisations, and labeling problems that arise in a variety of computer vision tasks, including for instance those derived from the generalised Potts model. Starting from the popular formulation of labeling as a convex relaxation by functional lifting, we show that FEM discretisation is valid for the most general case, where the regulariser is anisotropic and non-metric. While our findings are generic and applicable to different vision problems, we demonstrate their practical implementation in the context of semantic 3D reconstruction, where such regularisers have proved particularly beneficial. The proposed FEM approach leads to a smaller memory footprint as well as faster computation, and it constitues a very simple way to enable variable, adaptive resolution within the same model.

1 Introduction

A number of computer vision tasks, such as segmentation, multiview reconstruction, stitching and inpainting, can be formulated as *multi-label problems on continuous domains*, by functional lifting [7, 10, 24, 30, 32]. A recent example is semantic 3D reconstruction (*e.g.* [3, 15]), which solves the following problem: Given a set of images of a scene, reconstruct both its 3D shape and a segmentation into semantic object classes. The task is particularly challenging, because the evidence is irregularly distributed in the 3D domain; but it also possesses a rich, anisotropic prior structure that can be exploited. Jointly reasoning about shape and class allows one to take into account class-specific shape priors (*e.g.*, building walls should be smooth and vertical, and vice versa smooth, vertical surfaces are likely to be building walls), leading to improved reconstruction results. So far, models for the mentioned multi-label problems, and in particular for semantic 3D reconstruction, have been limited to axis-aligned discretisations. Unless the scenes are aligned with the coordinate

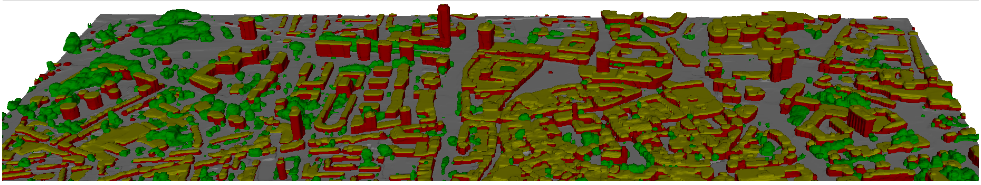

Figure 1: Semantic 3D model, estimated from aerial views with our FEM method.

axes, this leads to an unnecessarily large number of elements. Moreover, since the evidence is (inevitably) distributed unevenly in 3D, it also causes biased reconstructions. Thus, it is desirable to adapt the discretisation to the scene content (as often done for purely geometric surface reconstruction, *e.g.* [21]).

Our formulation makes it possible to employ a finer tesselation in regions that are likely to contain a surface, exploiting the fact that both high spatial resolution and high numerical precision are only required in those regions. Our discretisation scheme leads to a smaller memory footprint and faster computation, and it constitues a very simple technique to allow for arbitrary adaptive resolution levels within the same problem. *I.e.*, we can refine or coarsen the discretisation as appropriate, to adapt to the scene to be reconstructed. While our scheme is applicable to a whole family of finite element bases, we investigate two particularly interesting cases: Lagrange (P1) and Raviart-Thomas elements of first order. We further show that the grid-based voxel discretisation is a special case of our P1 basis, such that minimum energy solutions of "identical" discretisations (same vertex set) are equivalent.

2 Related Work

Since the seminal work [11] volumetric reconstruction from image data has evolved remarkably [9, 13, 17, 18, 19, 25, 39, 42]. Most methods use depth maps or 2.5D range scans for evidence [46, 47], represent the scene via an indicator or signed distance function in the volumetric domain, and extract the surface as its zero level set, *e.g.*, [26, 38].

Joint estimation of geometry and semantic labels, which had earlier been attempted only for single depth maps [22], has recently emerged as a powerful extension of volumetric 3D reconstruction from multiple views [1, 3, 15, 20, 35, 40, 41]. A common trait of these works is the integration of depth estimates and appearance-based labeling information from multiple images, with class-specific regularisation via shape priors.

Multi-label problems are in general NP-hard, but under certain conditions on the pairwise interactions, the original non-convex problem can be converted into a convex one via functional lifting and subsequent relaxation, *e.g.* [7]. This construction was further extended to anisotropic (direction-dependent) regularisers [37]. Moreover, [48] also relaxed the requirement that the regulariser forms a metric on the label set, yet its construction can only be applied after discretisation [24]. In this paper, we consider the relaxation in its most general form [48], but are not restricted to it. The latter construction is also the basis to the model of [15], whose energy model we adapt for our semantic 3D reconstruction method. Their voxel-based formulation can be seen as a special case of our discretisation scheme.

For (non-semantic) surface reconstruction, several authors prefer a data-dependent discretisation, normally a Delaunay tetrahedralisation of a 3D point cloud [16, 21, 43]. The occupancy states of the tetrahedra are found by discrete (binary) labeling, and the final surface is composed of the triangles that separate different labels. Loosely speaking, our proposed methodology can be seen either as an extension of [15] to arbitrary simplex partitions of the domain; or as an extension of [21] to semantic (multi-label) reconstruction.

We note that regular voxel partitioning of the volume leads to a high memory consumption and computation time. Yet, we are essentially reconstructing a 2D manifold in 3D space, and this can be exploited to reduce run-time and memory footprint. [20] use an octree instead of equally sized voxels to adapt to the local density of the input data. [3] go one step further and propose an adaptive octree, where the discretisation is refined on-the-fly, during optimisation. In our framework the energy is independent of the discretisation, it can thus be combined directly with such an adaptive procedure.

Also volumetric fusion via signed distance functions [28] benefits from irregular tesselations of 3D space, *e.g.*, octrees [36] or hashmaps [29]. In contrast to our work, these target real-time reconstruction and refrain from global optimisation, instead locally fusing depth maps. Their input normally is a densely sampled, overcomplete RGB-D video-stream, whereas we deal with noisy and incomplete inputs. To achieve high-quality reconstructions in our setting, we incorporate semantic information, leading to a multi-label problem.

Our work is based on the finite element method (FEM), *e.g.* [5, 33]. Introduced by Ritz [34] more than a century ago, and refined by Galerkin and Courant [8], FEM serves to numerically solve variational problems, by partitioning the domain into finite, parametrised elements. In computer vision FEM has been applied in the context of level-set methods [45] and for Total Variation [2]. To our knowledge, we are the first to apply it to multi-labeling.

3 Method

The multi-labeling problem [7, 24, 37, 48] in the domain $\Omega \subset \mathbb{R}^d$ is defined by finding m labeling functions $x^i : \Omega \to \{0,1\}, i = 1 \dots m$ as the solution of:

$$\inf_{x^i} \sum_{i=1}^{m} \int_{\Omega} \rho^i(z) x^i(z) \mathrm{d}z + J(x^i), \quad \text{s. t. } \sum_{i=1}^{m} x^i(z) = 1 \; \forall z \in \Omega, \tag{1}$$

where ρ models the data term for a specific label at location $z \in \Omega$ and J denotes a convex regularisation functional that enforces the spatial consistency of the labels. One prominent example is to chose $J := \|\cdot\|_2$, known as Total Variation, which penalises the perimeter of the individual regions [7, 30]. Note that in the two-label case (Potts model), this relaxation is exact after thresholding with any threshold from the open unit interval [7]. Although we are ultimately interested in non-metric regularisation, we start with the continuous, anisotropic model [37], and postpone the extension to the non-metric case to Sec. 3.5.

3.1 Convex Relaxation

The continuous model allows for an anisotropic regulariser in J: label transitions can be penalised on the area of the shared surface, as well as on the surface normal direction. This is achieved with problem-specific 1-homogeneous functions that emerge from convex sets, so called Wulff-shapes. A relaxation of $x^i(z) \in \{0,1\}$ to $x^i(z) \in [0,1]$ then leads to a convex energy, which can be written as the following saddle point problem, with primal functions x and dual functions λ:

$$\min_{x^i} \max_{\lambda^i} \sum_i \int_{\Omega} \rho^i(z) x^i(z) + \langle x^i(z), \nabla \cdot \lambda^i(z) \rangle \mathrm{d}z, \text{ s.t. } \lambda^i(z) - \lambda^j(z) \in W^{ij}, \sum_{i=1}^{m} x^i(z) = 1, x^i(z) \geq 0. \tag{2}$$

The constraints have to be fulfilled for all $z \in \Omega$. In addition to the primal variables x^i, we have introduced the dual vector-field $\lambda^i : \mathbb{R}^d \to \mathbb{R}^d$, whose pairwise differences are constrained to lie in the convex sets (Wulff-shapes) W^{ij}. By letting these shapes take an anisotropic form, one can then encode scene structure, *e.g.* [15, 37]. For our problem we demand Neumann conditions at the boundary of Ω, *i.e.* $\langle \lambda^i, \nu \rangle = 0, \forall z \in \partial\Omega$, because the scene will continue beyond our domain (ν is the normal of the domain boundary $\partial\Omega$).

3.2 Finite Element Spaces

Here, we can only informally introduce the basic idea of FEM and explain its suitability for problems of the form (2). We refer to textbooks [5, 12, 23] for a deeper and formal treatment.

One way to solve (2) is to approximate it at a finite number of regular grid points, using finite differences. FEM instead searches for a solution in a finite-dimensional vector space; this *trial* space is a subspace of the space in which the exact solution is defined. To that end, one chooses a suitable basis for the *trial* space, with basis functions of finite support, as well as an appropriate *test* function space. FEM methods then find approximate solutions to variational problems by identifying the element from the *trial* space that is orthogonal to all functions of the *test* function space. For our saddle-point problem, we can instead identify the *trial* space with our primal function space and the *test* space with its dual counterpart. Now, we can apply the same principles, and after discretisation our solution corresponds to the continuous solution defined by the respective basis. As (2) is already a relaxation of the original problem (1), we do not present an analysis of convergence at this point. Instead, the reader is referred to [2] for an introduction to this somewhat involved topic.

In order to choose a space with good approximation properties and suitable basis functions, we tesselate our domain into simplices. More formally, we define $M = \{F, V, S\}$ to be a simplex mesh with vertices $v \in V, v \in \mathbb{R}^d$, faces $f \in F$ defined by d, and simplices $s \in S$, defined by $d+1$ vertices that partition Ω: $\cup_k s_k = \Omega, s_l \cap s_k = f_{l,k} \in F$ – *i.e.* two adjacent simplices share only a single face. In this work, for a specific set of vertices V, we select M to be the corresponding Delaunay tetrahedralisation of Ω and only consider explicit bases. In particular, we focus on the Lagrangian (P1) basis, which we use in the following to derive our framework; and on the Raviart-Thomas (RT) basis. Details for the latter are given in the supplementary material. The main difference between them is that P1 leads to piecewise linear solutions, which must be thresholded, while RT leads to a constant labeling function per simplex, similar to discrete MAP solutions on CRFs. We note that constant labeling can lead to artefacts, such that the adaptiveness of the FEM model becomes even more important.

The idea of both derivations is similar: *(i)* select a basis for our primal (P1) or dual (RT) variable set, *(ii)* find a suitable form via the divergence theorem and Fenchel duality, *(iii)* extend to the non-metric case, following a principle we term ”label mass preservation”.

3.3 Lagrange Elements

The Lagrange $\mathrm{P}^k(M)$ basis functions describe a *conforming* polynomial basis of order $k+1$ on our simplex mesh M, *i.e.* its elements belong to the Hilbert space of differentiable function with finite Lebesgue measure on the domain Ω: $\mathrm{P}^k(M) \subset H^1(\Omega) := \{p \in L^2(\Omega), \nabla p \in (L^2(\Omega)^d)\}$. We are interested in the Lagrange basis of first order, $\mathrm{P}^1(M)$:

$$P^1(M) := \{p : \Omega \to \mathbb{R} | p \in C(\Omega),\, p(x) := \sum_{s \in S} c_s^\mathsf{T} x + d_s, c_s \in \mathbb{R}^d, d_s \in \mathbb{R}, \text{ if } x \in s \text{ and } 0 \text{ else}\}. \quad (3)$$

We construct our linear basis with functions that are defined for each vertex v of a simplex s and can be described in local form with barycentric coordinates:

$$p^1_{s,v}(x) := \alpha_v \text{ with } x = \sum_{v \in s} \alpha_v v,\ \sum_{v \in s} \alpha_v = 1,\ \alpha_v \geq 0 \quad \text{if } x \in s \text{ and } 0 \text{ else.} \quad (4)$$

In each simplex, one can define a scalar field $\phi_s(x) \in \mathbb{R}$ and compute a gradient in this basis that will be constant per simplex s (*cf.* Fig. 2):

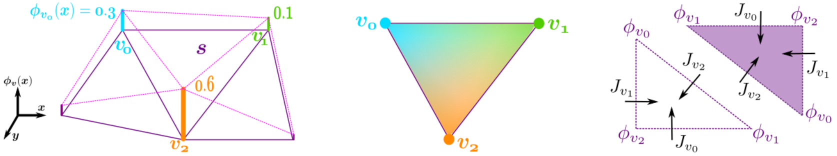

Figure 2: *Left:* Illustration of P1 basis function shape. *Middle:* Scalar field defined as a convex combination of basis coefficients. *Right:* Gradient definition in a simplex (5).

$$\phi_s(x) := \sum_{v\in s} \phi_v p^1_{s,v} \text{ and } \nabla\phi_s = \sum_{v\in s} \phi_v J_v, \tag{5}$$

with coefficients $\phi_v \in \mathbb{R}$. $J_v \in \mathbb{R}^d$ denotes a vector that is normal to the face f_v opposite node v, has length $\frac{|f_v|}{|s|d}$, and points towards the simplex centre. $|f_v|$ denotes the area of the face f_v, and $|s|$ the volume of the simplex s (*cf*. Fig. 2 and supplementary material).

3.4 Discretisation

To apply our Lagrange basis to (2) we first make use of the divergence theorem:

$$\int_\Omega \langle x^i(z), \nabla\cdot\lambda^i(z)\rangle \mathrm{d}z = \int_\Omega \langle \nabla x^i(z), \lambda^i(z)\rangle \mathrm{d}z - \underbrace{\int_{\partial\Omega} \langle \nu(z), \lambda^i(z)\rangle \mathrm{d}z}_{=0}. \tag{6}$$

The latter summand vanishes by our choice of λ. Our approach for a discretisation in the Lagrange basis is to choose the labeling function $x^i \in P^1(M)$. This implies that our dual space consists of constant vector-fields per simplex: $\lambda^i_s \in \mathbb{R}^d$. To fulfill the constraint set in (2) we have to verify that, per simplex, the λ^i_s lie in the respective Wulff-shape. The simplex constraints on the x^i have to be modeled per vertex. According to (4), the labeling functions are convex combinations of their values at the vertices and thus stay within the simplex.

We also have to convert the continuous data costs ρ^i into a cost per vertex ρ^i_v, which can be achieved by convolving the continuous cost with the respective basis function: $\rho^i_v := \int_\Omega \sum_{s\in\mathcal{N}(v)} \phi_s(x)\rho^i(x)\mathrm{d}x$. In practice, the integral can be computed by sampling ρ. Integrating the right hand side in (6) over the simplex s leads to a weighting with its volume $|s|$ and the energy (2) in the discrete setting becomes:

$$\min_{x^i}\max_{\lambda^i} \sum_{v,i} \rho^i_v x^i_v + \sum_{s,i} |s| \langle \nabla x^i, \lambda^i_s\rangle \text{ s.t. } (\lambda^i_s - \lambda^j_s) \in W^{ij}\ \forall i<j,\ s\in S, \sum_{i=1}^m x^i_v = 1,\ x^i_v \geq 0\ \forall v \in V. \tag{7}$$

3.5 Non-metric extension

To start with, we note that a non-metric model does not exist in the continuous case [27] and our extension works only *after* the discretisation into the FEM basis. Please refer to the supplementary material for an in-depth discussion. Note that our label set of semantic classes does not have a natural order (in contrast to, *e.g.*, stereo depth or denoised brightness); and also the direction-dependent regulariser is unordered and does not induce a metric cost. To allow for non-metric regularisation we transform the constraint set ($\lambda^i - \lambda^j \in W^{ij}$), by introducing auxiliary variables z^{ij} and Lagrange multipliers y^{ij}, and use Fenchel-Duality:

$$\max_{\lambda^i_s, z^{ij}_s} \min_{y^{ij}_s} \sum_{i<j} \langle (\lambda^i_s - \lambda^j_s) - z^{ij}, y^{ij}_s \rangle - \delta_{W^{ij}}(z^{ij}_s) = \max_{\lambda^i_s} \min_{y^{ij}_s} \sum_{i<j} -\langle (\lambda^i_s - \lambda^j_s), y^{ij}_s \rangle + ||y^{ij}_s||_{W^{ij}}. \tag{8}$$

The dual functions of the indicator functions for the convex sets W^{ij} are 1-homogeneous, of the form $||\cdot||_{W^{ij}} := \sup_{w\in W^{ij}} w^\mathsf{T}\cdot$. Recall that our label costs are not metric: $\forall i<j<k:$

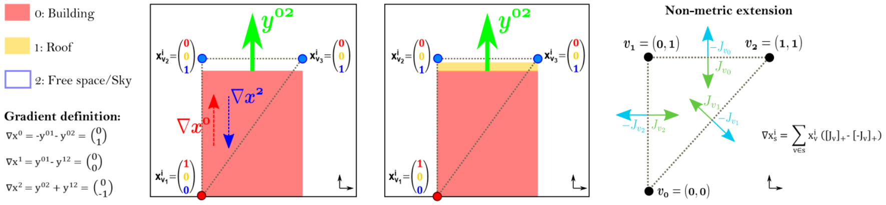
0: Building
1: Roof
2: Free space/Sky
Gradient definition:
$\nabla x^0 = -y^{01} - y^{02} = \binom{0}{1}$
$\nabla x^1 = y^{01} - y^{12} = \binom{0}{0}$
$\nabla x^2 = y^{02} + y^{12} = \binom{0}{-1}$
y^{02}
∇x^0
∇x^2
$x^i_{v_1} = \begin{pmatrix}1\\0\\0\end{pmatrix}$
$x^i_{v_2} = \begin{pmatrix}0\\0\\1\end{pmatrix}$
$x^i_{v_3} = \begin{pmatrix}0\\0\\1\end{pmatrix}$
Non-metric extension
$v_1 = (0,1)$
$v_2 = (1,1)$
$v_0 = (0,0)$
$-J_{v_0}$
J_{v_0}
J_{v_1}
$-J_{v_1}$
$-J_{v_2}$
J_{v_2}
$\nabla x^i_s = \sum_{v \in s} x^i_v ([J_v]_+ - [-J_v]_+)$

Figure 3: *Left:* Without our non-metric extension, optimisation w.r.t. (7) can lower transition costs by inserting another label (here 1 between 0 and 2). *Right:* A solution is to split the gradients of the indicator functions and use direction-dependent variables x^{ij}.

$|y^{ij}|_{W^{ij}} + |y^{jk}|_{W^{jk}} \geq |y^{ik}|_{W^{ik}}$, does not hold. It was shown [48] that a regulariser of the form (8) transforms any non-metric cost to the metric case. Figure 3 shows an example. Here, an expensive transition between labels 0 and 2 will be replaced by two cheaper transitions 0–1 and 1–2. To prevent this, we replace the y^{ij} with direction dependent variables x^{ij}: We rearrange (8) and combine the first summand with the regulariser from (7) to arrive at the following equations (for now ignoring $|s|$):

$$\sum_s \sum_i \langle \nabla x^i, \lambda^i_s \rangle - \sum_i \langle \lambda^i_s, \sum_{j \neq i} \left(y^{ij}_s [i < j] - y^{ji}_s [i > j] \right) \rangle,$$

with $[\cdot]$ denoting Iverson brackets. Let $x^{ij} := [y^{ij}]_+$ and $x^{ji} := [-y^{ji}]_+$, where $[\cdot]_+ := \max(0, \cdot)$. Expanding the gradient (5) we get, per simplex s,

$$\sum_i \sum_{v \in s} \lambda^i_s x^i_v ([J_v]_+ - [-J_v]_+) - \langle \lambda^i_s, \sum_{j : i \neq j} (x^{ij} - x^{ji}) \rangle, \tag{9}$$

which we analyse further to achieve non-metric costs. It was observed in [48] that the $x^{ij} \in \mathbb{R}^d$ can be interpreted as encoding the "label mass" that transitions from label i to label j in a specific direction. Positivity constraints (by definition) on the x^{ij} avoid the transport of negative label mass. To anchor transport of mass on the actual mass of label i present at a vertex, we introduce the variables x^{ii} for the mass that remains at label i, and split the above constraints into two separate sets with the help of additional dual variables θ:

$$\lambda^i_s (\sum_{v \in s} x^i_v [J_v]_+ - \sum_j x^{ij}_s) + \theta^i_s (\sum_{v \in s} x^i_v [-J_v]_+ - \sum_j x^{ji}_s) + \sum_{i,j} \delta_{\geq 0}(x^{ij}_s). \tag{10}$$

Note that this construction is only possible because our elements (simplices) are of strictly positive volume, in contrast to zero sets in Ω w.r.t. the Lebesgue measure. Finally, we can write down our discrete energy in the Lagrange basis defined on the simplex mesh M:

$$\begin{aligned} \min_{x^i, x^{ij}} \max_{\lambda^i, \theta^i} & \sum_{v \in V} \sum_i \rho^i_v x^i_v + \delta_\Delta(x^i_v) + \sum_{i<j} \sum_{s \in S} |s| ||x^{ij}_s - x^{ji}_s||_{W^{ij}} + \\ & \sum_s \sum_i \theta^i_s (\sum_{v \in s} x^i_v [-J_v]_+ - \sum_j x^{ji}_s) + \sum_s \sum_i \lambda^i_s (\sum_{v \in s} x^i_v [J_v]_+ - \sum_j x^{ij}_s) + \sum_{i,j} \delta_{\geq 0}(x^{ij}_s), \end{aligned} \tag{11}$$

where we have moved the weighting with $|s|$ from the constraint set to the regulariser, and denote by $\delta_\Delta(\cdot)$ the indicator function of the unit simplex.

4 Semantic Reconstruction Model

A prime application scenario for our FEM multi-label energy model (11) is 3D semantic reconstruction. In particular, we focus on an urban scenario and let our labeling functions

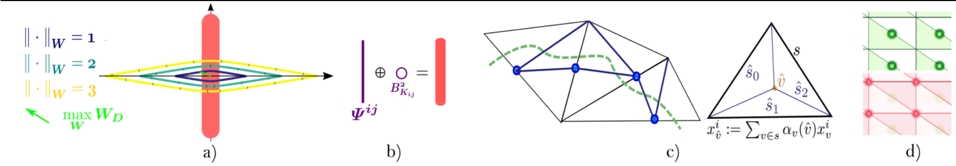

Figure 4: *(a):* Wulff-shape (*red*) with isolines. *(b):* Minkowski sum of two Wulff-shapes. *(c):* Simplices are split after inserting new vertices (blue) close to the surface (green). *Right:* Initialisation of vertices after refinement. *(d):* Finite differences on a regular grid ([15]) only cover constraints in *green* areas, the P1 basis covers all of the domain Ω.

encode *freespace* ($i = 1$), *building wall*, *roof*, *vegetation* or *ground*. Objects that are not explicitly modeled are collected in an extra *clutter* class. We define the data cost ρ at a 3D-point $x \in \Omega$ as in [15]: project x into the camera views $c \in \mathcal{C}$, and retrieve the corresponding depth $\hat{d}_c(x)$ and class likelihoods $\sigma^i_c(x)$ from the image space. The σ^i are obtained from a MultiBoost classifier. For the depth we look at the difference between the actual distance $d_c(x)$ to the camera and the observed depth: $d(x,c) := d_c(x) - \hat{d}_c(x)$. For the *freespace* label we always set the cost to 0, for $i \neq 1$ we define:

$$\rho^i(x) := \sum_{c \in \mathcal{C}} \sigma^i_c(x)[(k-1)\varepsilon \leq d(x,c) \leq k\varepsilon] + \beta[|d(x,c)| \leq k\varepsilon]\,\mathrm{sign}(d(x,c)). \tag{12}$$

This model assumes independence of the per-pixel observations, and exponentially distributed inlier noise in the depthmaps, bounded by a parameter $k\varepsilon$ (k=3 in practice). It is essentially a continuous version of [15], see that paper for details. The parameter ε sets a lower bound for the minimal height of the simplices in the tesselation, and thus defines the target resolution. The discretisation of the data cost involves a convolution with the respective basis functions, which can be approximated via sampling. Please refer to the supplementary material for details. The Wulff-shapes W^{ij} in (11) are given as the Minkowski sum of the L_2-Ball, $B^2_{\kappa^{ij}} := \{x \in \mathbb{R}^d \,|\, \|x\|_2 \leq \kappa^{ij}\}$ and an anisotropic shape Ψ^{ij}: $W^{ij} := \Psi^{ij} \oplus B^2_{\kappa^{ij}}$. In the isotropic part, κ^{ij} contains the neighbourhood statistics of the classes. The anisotropic part Ψ^{ij} models the likelihood of a transition between classes i and j in a certain direction. Fig. 4 (a,b) shows an example. For our case we prefer flat, horizontal surfaces at the following label transitions: *ground-freespace*, *ground-building*, *building-roof*, *ground-vegetation* and *roof-freespace*. A second prior prefers vertical boundaries for the transitions *building-freespace* and *building-vegetation*. More details on the exact form can be found in [15].

The energy (11) is already in primal-dual form, such that we can apply the minimisation scheme of [6], with pre-conditioning [31]. That numerical scheme requires us to project onto shapes that are Minkowski sums of convex sets. In our case, the sets are simple and the projection onto each shape can be performed in closed form. We employ a Dykstra-like projection scheme [4], which avoids storing additional variables and proves remarkably efficient, see supplementary material. We also project the labeling functions x^i directly onto the unit simplex [44]. In order to extract the transition surface, we employ a variant of marching tetrahedra (triangles) [38], using the isolevel at 0.5 for each label.

We conclude with two interesting remarks. First, note that a tesselation with a regular grid [15] can be seen as a simplified version of our discretisation in the P^1 Lagrange basis. In Fig. 4d we consider the 2D case of the regular grid used in [15]. Here, variables are defined at voxel level. In its dual graph, the vertex set consists of the corners of the primal grid cubes, leading to shifted indicator variables. Per vertex the data term is mainly influenced from the cost in its Voronoi area. Similarly, [15] evaluates the data cost at grid centers, approximately

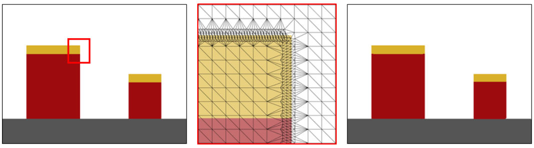

Figure 5: *Left:* Synthetic 2D scene, colors indicate ground (*gray*), building (*red*) and roof (*yellow*). *Middle:* Control mesh. *Right:* Example reconstruction.

overall acc. [%]	Tetra	Octree	MB
Scene 1	84.0	83.9	82.5
Scene 2	92.5	92.8	89

Table 1: Quantitative comparison with octree model [3] and MultiBoost input data.

corresponding to integration within the respective Voronoi-area of grid cell. Furthermore, taking finite differences in this regular case corresponds to verifying constraints for only one of the two triangles (Fig. 4d). The supplementary material includes a more formal analysis.

Second, our formulation is adaptive, in the sense of [3]: Hierarchical refinement of the tesselation can only decrease our energy. Hence, our scheme is applicable when refining the model on-the-fly. We again must defer a formal proof to the supplementary material and give an intuitive, visual explanation in Fig. 4c . Assume that $(x^*, \lambda^*, \theta^*)$ is an optimal solution for a certain triplet $M = \{F, V, S\}$. Then a refined tesselation $\hat{M} = \{\hat{F}, \hat{V}, \hat{S}\}$ can be found by introducing additional vertices, *i.e.* $V \subset \hat{V}$ (ideally on the label transition surfaces). To define a new set of simplices, we demand that no faces are flipped, $\forall \hat{s} \in \hat{S}, \exists s \in S$ with $\hat{s} \cap s = \hat{s}$. Then one can find a new variable set and data cost $\hat{\rho}$ with the same energy: We initialise the new variables from the continuous solution at the respective location, and find new $\rho_{\hat{v}}$ by integration. Subsequent minimisation in the refined mesh can only decrease the energy. The argument works in both ways: Vertices that have the same solution as their adjacent neighbors can be removed without changing the energy. For now we stick to this simple scheme, future work might explore more sophisticated ideas, *e.g.* along the lines of [14].

5 Evaluation

Before we present results on challenging real 3D data we evaluate our method in 2D on a synthetic dataset. All results are obtained with a multi-core implementation, on a 12-core, 3.5 GHz machine with 64GB RAM. For clarity, we only present the Lagrange discretisation. We refer to the supplementary material for an evaluation of the Raviart-Thomas discretisation.

Input Data. We create a synthetic 2D scene composed of 4 labels: *free space, building, ground* and *roof*, surrounded by 17 virtual cameras. To replace depth maps and class-likelihood images, we extract 2D points on the boundary "surface" and assign ground truth label costs to each point. For the evaluation in 3D, we use three real-world aerial mapping data sets. Our method requires two types of input data: depthmaps and pixel-wise class probabilities (*cf*. Sec. 4). Moreover, we build a *control mesh M* around the initially predicted surface, to facilitate our FEM discretisation. Ideally, the *control mesh* enwraps the true surface, using a finer meshing close to it. We densely evaluate the data cost at the vertices of a regular *data cost grid* and let each *control vertex* accumulate the cost of its nearest neighbours in that grid, to approximate an integration over its Voronoi cell.

2D Lagrange results. Fig. 5 illustrates the result we obtain in a *perfect* setting. The original 2D image serves as ground truth for our quantitative evaluation. In this baseline setting, our method achieves 99.8% of *overall accuracy* and 99.7% of *average accuracy*, confirming the soundness of our Lagrange discretisation. In order to evaluate how our model behaves in a more realistic setting, we conduct a series of experiments where we incrementally add different types of perturbations. Our algorithm is tested against: *(i)* noise in the initial 2D point cloud, respectively depth maps, *(ii)* wrong class probabilities and *(iii)* ambiguous class

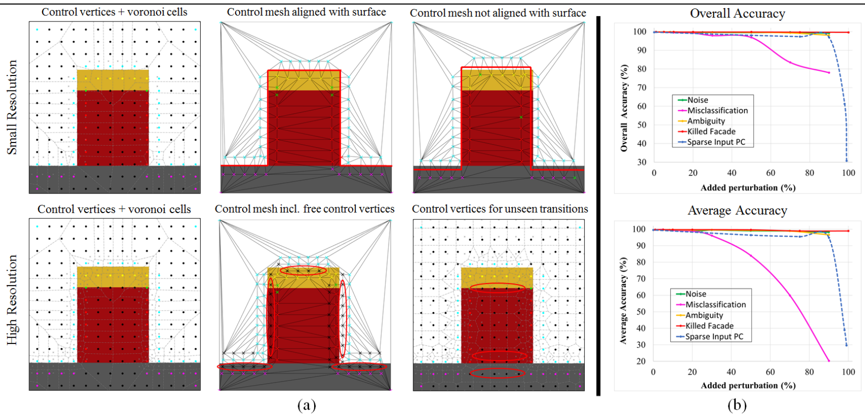

Figure 6: (a) Illustration of the control mesh foundation. Dots represent values of the datacost grid and crosses the *control vertices*. Voronoi cells of the *control vertices* are depicted with dashed grey lines and the *control mesh* with a solid black line. Colors indicate ground (*purple*), building (*red*), roof (*yellow*), free space (*cyan*) and no datacost (*black*). (b) Quantitative evaluation of Lagrange FEM method w.r.t. different degradations of the input data.

probabilities of random subsets, *(iv)* missing data, *e.g.* deleting part of a facade to simulate unobserved areas, *(v)* sparsity of the initial point cloud. Fig. 6b illustrates the influence of defective inputs. Under reasonable assumptions on the magnitude of the investigated perturbations, we do not observe a significant loss in accuracy. The reconstruction quality starts to decrease if more than half of the input data is misclassified or if the input point cloud is excessively sparse, meaning that >50% of the input is wrong or nonexistent. Average accuracy is naturally more sensitive, due to the larger relative error in small classes.

Influence of the control mesh. Recall from (12) that the data cost of a *control vertex* $v \in V$ is approximately equal to an integral of ρ over its respective Voronoi area (*cf*. Fig. 6a, left). Therefore, but also because of the sign change in (12), vertices close to the surface receive small cost values and are mainly steered by the regulariser, *i.e.* these vertices realise a smooth surface. On the other hand, vertices that integrate only over areas with positive or negative sign determine the inside/outside decision, but are more or less agnostic about the exact location of the surface. We conclude that a sufficient amount of *control vertices* should lie within the band $[\hat{d}-3\varepsilon;\hat{d}+3\varepsilon]$ defined by the truncation of the cost function around the observed depth $\hat{d}$ (*cf*. (12) and [15]). Ideally *control vertices* are equally distributed along each line-of-sight in front and behind the putative match (*cf*. Fig. 6a, middle column). Undersampling within the near-band can lead to smooth, but inaccurate results (*cf*. Fig. 6a, top right). Unobserved transitions, *e.g. building-ground* or *roof-building*, can also lead to problems if the affected simplices are too large. To mitigate the effect, we add a few vertices (*e.g.*, a sparse regular grid) on top of the *control mesh* (*cf*. Fig. 6a, bottom row). Finally, oversampling each line-of-sight in order to increase the resolution of the *control mesh* is not recommended, the right spacing is determined by the noise level and ε and k, chosen in (12).

To conclude, it is an important advantage of the FEM framework that additional vertices can be inserted as required, without changing the energy. In future work we will use this flexibility to develop smarter *control meshes*, possibly as a function of the *local* noise level.

3D Lagrange results. To test our algorithm on real world data, we focus on a dataset from the city of Enschede. Complementary results for other datasets are shown in the supplementary material. As baseline we use [3], the current state-of-the-art in large-scale semantic

3D reconstruction. Due to the lack of 3D ground truth, we follow their evaluation protocol and back-project subsets of the 3D reconstruction to image space, where it is compared to a manual segmentation. As can be seen in Fig. 7 and Tab. 1, the two results are similar in terms of quantitative correctness. We note that measuring labeling accuracy in the 2D projection does not consider the geometric quality of the reconstruction within regions of a single label.

Figs. 1 and 7 show city-modelling results obtained from (nadir and oblique) aerial images. Visually, our models are crisper and less "clay-like". Compared to axis-aligned discretisation schemes, *e.g.* [3, 15], our method appears to better represent surfaces not aligned with the coordinate axis, and exhibit reduced grid aliasing. Both effects are consistent with the main strength of the FEM framework, to adapt the size *and* the orientation of the volume elements to the data. Small tetrahedra, and vertices that coincide with accurate 3D points on surface discontinuities, favour sharp surface details and crease edges (*e.g.*, substructures on roofs). Faces that follow the data points rather than arbitrary grid directions mitigate aliasing on surfaces not aligned with the coordinate axes (*e.g.*, building walls). The freedom of a local *control mesh* unleashes the power of the regulariser in regions where the evidence is weak or ambiguous (*e.g.*, roads, weakly textured building parts).

As already mentioned, our FEM framework can be readily combined with on-the-fly adaptive computation, as used in the baseline [3]. Compared to their voxel/octree model, adaptive refinement is straight-forward, due to the flexibility of the FEM framework, which allows for the introduction of arbitrary new vertices. As a preliminary proof-of-concept, we have tested the naive refinement scheme described in Sec. 4. We execute three refinement steps, where we repeatedly reconstruct the scene and subsequently refine simplices that contain surface transitions, while lowering ε by half. Compared to computing everything at the final resolution, this already yields substantial savings of 89–97% in memory and 82–93% in computation time, without any loss in accuracy. Targetting $\varepsilon \geq \frac{1}{\sqrt{3}}$ (measured w.r.t. a bounding box of 256 units), the runtimes for the tested scenes are 1h04m–1h47m and memory consumption is 573–764 MB, on a single machine.

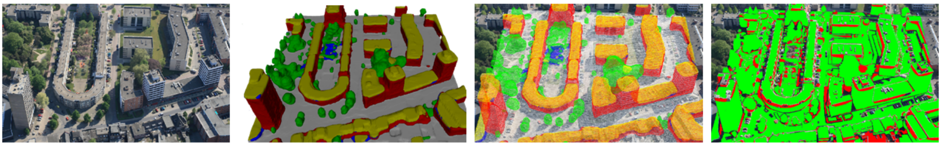

Figure 7: Quantitative evaluation of Scene 1 from Enschede. *Left:* One of the input images. *Middle left:* Semantic 3D model. *Middle right:* Back-projected labels overlayed on the image. *Right:* Error map, misclassified pixels are marked in red.

6 Conclusion

We have proposed a novel framework for the discretisation of multi-label problems, and have shown that, in the context of semantic 3D reconstruction, the increased flexibility of our scheme allows one to better adapt the discretisation to the data at hand. Our basic idea is generic and not limited to semantic 3D reconstruction or the specific class of regularisers. We would like to explore other applications where it may be useful to abandon grid discretisations and move to a decomposition into simplices.

Acknowledgements: Audrey Richard was supported by SNF grant 200021_157101. Christoph Vogel and Thomas Pock acknowledge support from the ERC starting grant 640156, 'HOMOVIS'.

References

[1] Yingze Bao, Manmohan Chandraker, Yuanqing Lin, and Silvio Savarese. Dense object reconstruction using semantic priors. In *CVPR*, 2013.

[2] Sören Bartels. Total variation minimization with finite elements: Convergence and iterative solution. *SIAM*, 50(3), 2012.

[3] 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.

[4] J. P. Boyle and R. L. Dykstra. A method for finding projections onto the intersection of convex sets in Hilbert spaces. *Lecture Notes in Statistics*, 1986.

[5] Franco Brezzi and Michel Fortin. *Mixed and Hybrid Finite Element Methods*. Springer-Verlag New York, 1991.

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

[7] Antonin Chambolle, Daniel Cremers, and Thomas Pock. A convex approach to minimal partitions. *SIAM*, 5(4), 2012.

[8] R. Courant. Variational methods for the solution of problems of equilibrium and vibrations. *Bull. Amer. Math. Soc.*, 49(1), 01 1943.

[9] D. Cremers and K. Kolev. Multiview stereo and silhouette consistency via convex functionals over convex domains. *PAMI*, 33(6), 2011.

[10] D. Cremers, T. Pock, K. Kolev, and A. Chambolle. Convex Relaxation Techniques for Segmentation, Stereo and Multiview Reconstruction. In *Advances in Markov Random Fields for Vision and Image Processing*. MIT Press, 2011.

[11] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. *SIGGRAPH*, 1996.

[12] Ricardo G. Durán. *Mixed Finite Element Methods*. Springer Berlin Heidelberg, 2008.

[13] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multi-view stereopsis. *PAMI*, 2010.

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

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

[16] M. Jancosek and T. Pajdla. Multi-view reconstruction preserving weakly-supported surfaces. In *CVPR*, 2011.

[17] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *EUROGRAPHICS*, 2006.

[18] K. Kolev, T. Brox, and D. Cremers. Fast joint estimation of silhouettes and dense 3D geometry from multiple images. *PAMI*, 2012.

[19] Ilya Kostrikov, Esther Horbert, and Bastian Leibe. Probabilistic labeling cost for high-accuracy multi-view reconstruction. In *CVPR*, 2014.

[20] Abhijit Kundu, Yin Li, Frank Dellaert, Fuxin Li, and James M. Rehg. Joint semantic segmentation and 3d reconstruction from monocular video. In *ECCV*, 2014.

[21] Patrick Labatut, Jean-Philippe Pons, and Renaud Keriven. Efficient Multi-View Reconstruction of Large-Scale Scenes using Interest Points, Delaunay Triangulation and Graph Cuts. In *ICCV*, 2007.

[22] L'ubor Ladický, Paul Sturgess, Christopher Russell, Sunando Sengupta, Yalin Bastanlar, William Clocksin, and Philip Torr. Joint optimisation for object class segmentation and dense stereo reconstruction. In *BMVC*, 2010.

[23] Mats G. Larson and Fredrik Bengzon. *The Finite Element Method: Theory, Implementation, and Applications*. Springer Publishing Company, Incorporated, 2013.

[24] Jan Lellmann and Christoph Schnörr. Continuous multiclass labeling approaches and algorithms. *SIIMS*, 4(4), 2011.

[25] Shubao Liu and David B. Cooper. Ray Markov random fields for image-based 3d modeling: Model and efficient inference. In *CVPR*, 2010.

[26] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH*, 1987.

[27] F. Maggi. *Sets of Finite Perimeter and Geometric Variational Problems: An Introduction to Geometric Measure Theory*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2012.

[28] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *ISMAR*, Washington, DC, USA, 2011.

[29] Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Marc Stamminger. Real-time 3d reconstruction at scale using voxel hashing. *ACM Trans. Graph.*, 32(6), November 2013.

[30] Claudia Nieuwenhuis, Eno Töppe, and Daniel Cremers. A survey and comparison of discrete and continuous multi-label optimization approaches for the potts model. *IJCV*, 104(3), 2013.

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

[32] Thomas Pock, Daniel Cremers, Horst Bischof, and Antonin Chambolle. Global solutions of variational models with convex regularization. *SIIMS*, 3(4), 2010.

[33] J. Reddy. *An Introduction to the Finite Element Method.* McGraw-Hill Education, 2005.

[34] Walter Ritz. Über eine neue methode zur lösung gewisser variationsprobleme der mathematischen physik. *Journal für die reine und angewandte Mathematik*, 1909.

[35] Nikolay Savinov, L'ubor Ladický, Christian Häne, and Marc Pollefeys. Discrete optimization of ray potentials for semantic 3d reconstruction. In *CVPR*, 2015.

[36] F. Steinbruecker, J. Sturm, and D. Cremers. Volumetric 3d mapping in real-time on a cpu. In *ICRA*, 2014.

[37] E. Strekalovskiy and D. Cremers. Generalized ordering constraints for multilabel optimization. In *ICCV*, 2011.

[38] G. Treece. Regularised marching tetrahedra: improved iso-surface extraction. *Computers & Graphics*, 23(4), August 1999. ISSN 00978493.

[39] Ali Osman Ulusoy, Michael J. Black, and Andreas Geiger. Patches, planes and probabilities: A non-local prior for volumetric 3D reconstruction. In *CVPR*, 2016.

[40] Ali Osman Ulusoy, Michael J. Black, and Andreas Geiger. Semantic multi-view stereo: Jointly estimating objects and voxels. In *CVPR*, 2017.

[41] Vibhav Vineet, Ondrej Miksik, Morten Lidegaard, Matthias Nießner, Stuart Golodetz, Victor A. Prisacariu, Olaf Kähler, David W. Murray, Shahram Izadi, Patrick Perez, and Philip H. S. Torr. Incremental dense semantic stereo fusion for large-scale semantic scene reconstruction. In *ICRA*, 2015.

[42] George Vogiatzis, Carlos Hernández Esteban, Philip H. S. Torr, and Roberto Cipolla. Multiview stereo via volumetric graph-cuts and occlusion robust photo-consistency. *PAMI*, 29(12), 2007.

[43] H. H. Vu, P. Labatut, J. P. Pons, and R. Keriven. High accuracy and visibility-consistent dense multiview stereo. *PAMI*, 34(5), May 2012.

[44] Weiran Wang and Miguel Á. Carreira-Perpiñán. Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application. *CoRR*, abs/1309.1541, 2013.

[45] Martin Weber, Andrew Blake, and Roberto Cipolla. Sparse finite elements for geodesic contours with level-sets. In Tomás Pajdla and Jiří Matas, editors, *ECCV*, 2004.

[46] C. Zach, T. Pock, and H. Bischof. A globally optimal algorithm for robust TV-L1 range image integration. In *ICCV*, 2007.

[47] Christopher Zach. Fast and high quality fusion of depth maps. *3DV*, 2008.

[48] Christopher Zach, Christian Häne, and Marc Pollefeys. What is optimized in convex relaxations for multilabel problems: Connecting discrete and continuously inspired MAP inference. *PAMI*, 2014.

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

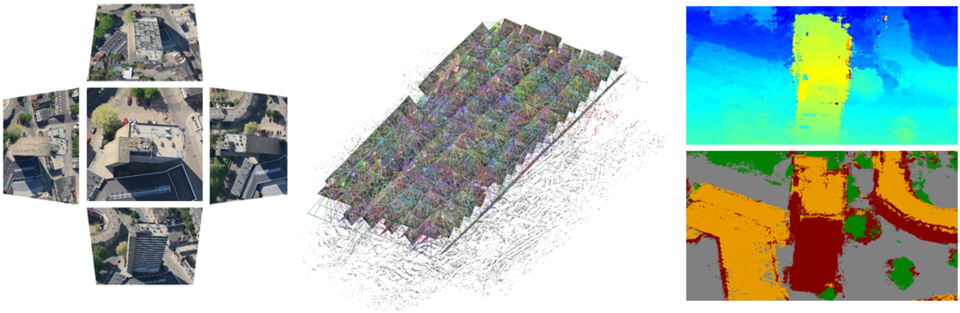

Figure 1: Input data. *Left:* aerial input images for one position (four oblique views to the north, east, south, west, and a nadir image). *Middle:* oriented image block. *Right:* depth map and class probabilities (visualised by maximum-likelihood labels).

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.

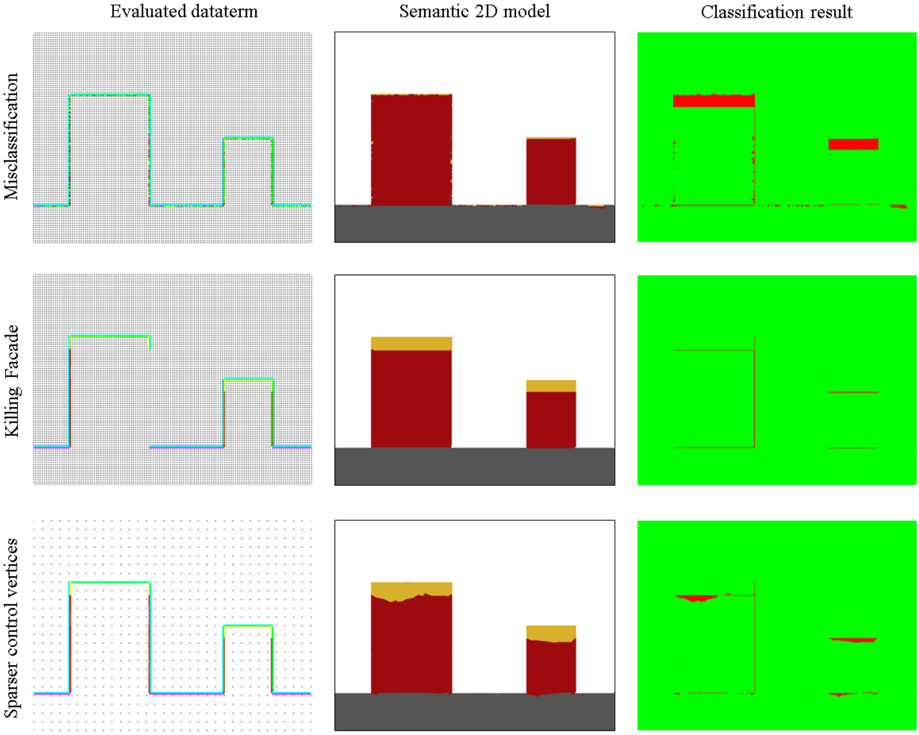

Figure 2: Example scenes of our 2D data set and results obtained with our Lagrange FEM method. *Left:* Data term at vertices of the *control mesh*. Colors for the data cost indicate: *free space/empty space* (cyan), *building* (red), *ground* (pink), *roof* (yellow), *occupied space* (green) and *no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

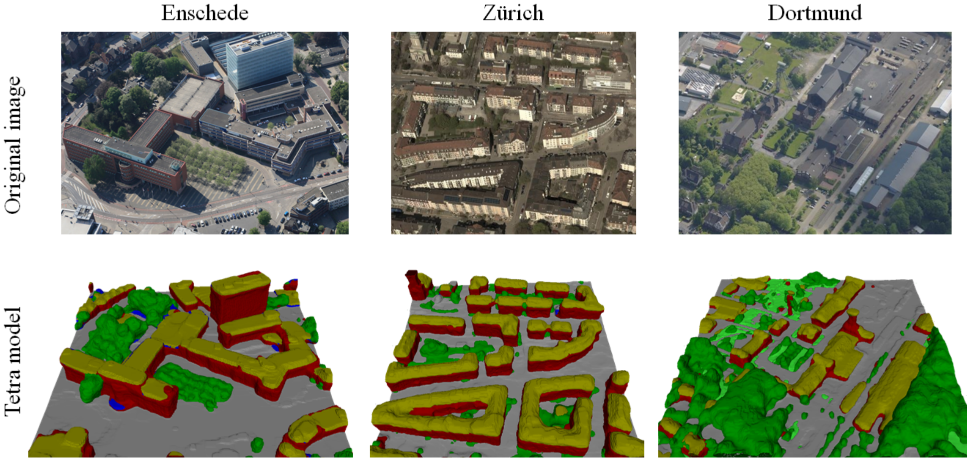

Figure 3: Additional datasets. *First row:* Original aerial images. *Second row:* Semantic 3D models obtained with the Lagrange FEM method for Enschede (left), Zürich (middle) and Dortmund (right). For the latter, light green denotes an additional class *grass and agricultural fields*.

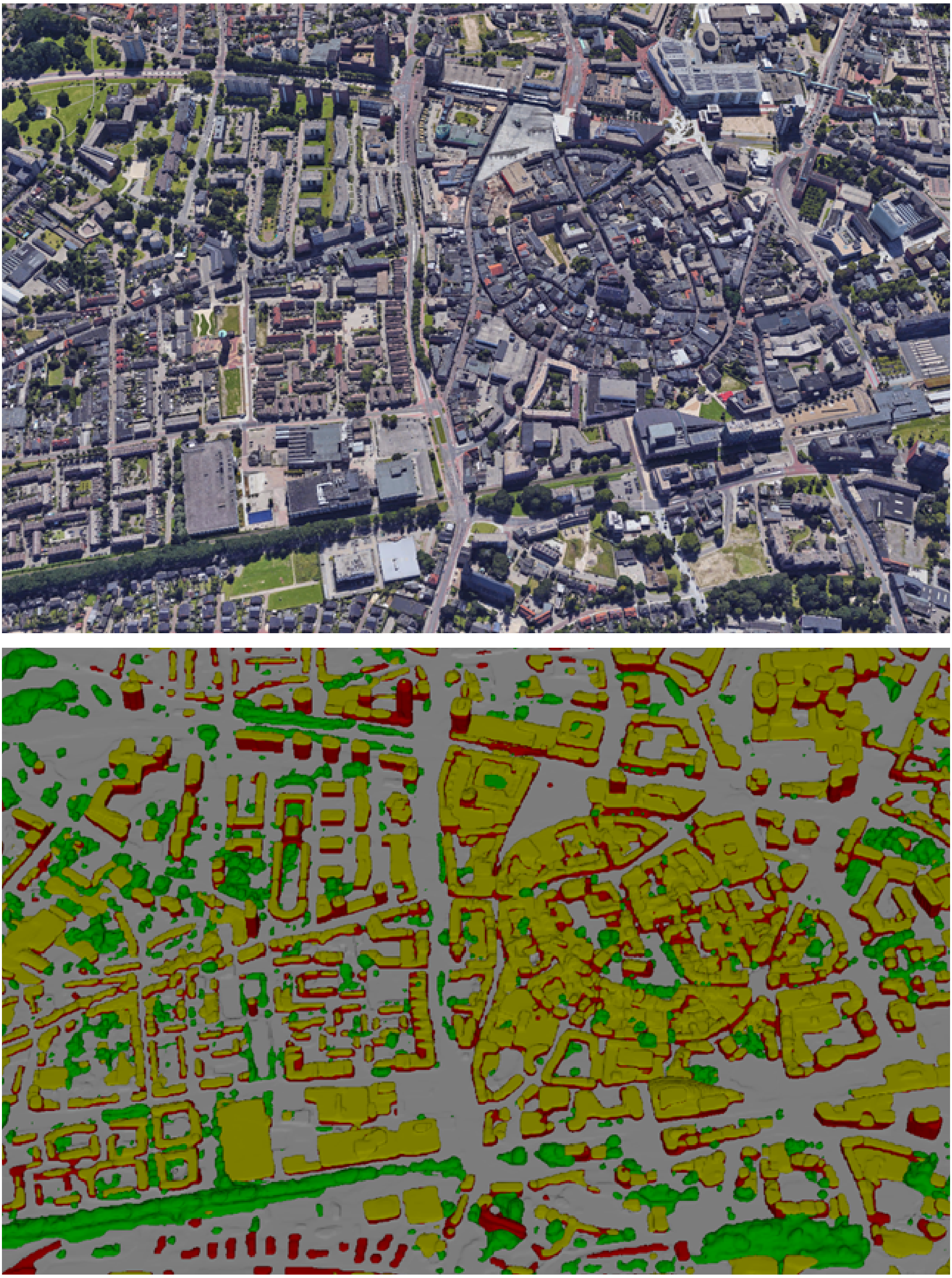

Figure 4: Large-scale semantic 3D reconstruction of Enschede (Netherlands), computed from aerial images with our Lagrange FEM method. *Top:* View from *Google Earth* (not used during reconstruction). *Bottom:* Our model from matching viewpoint.

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 $J_{v_l}, l = 1 \ldots d$ 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 $i = 1 \ldots m$, 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:

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

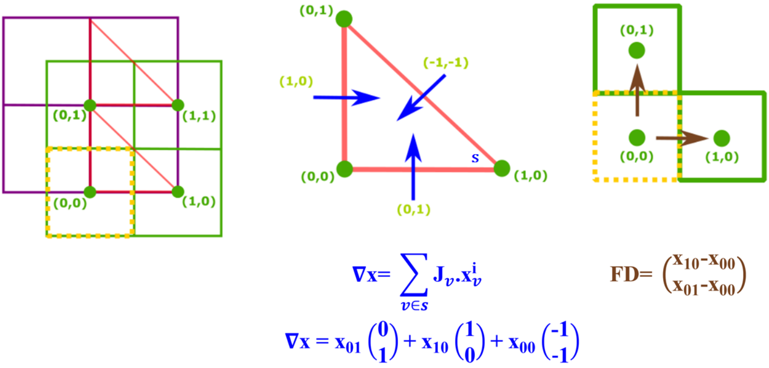
(0,1)
(1,1)
(0,0)
(1,0)
(0,1)
(-1,-1)
(1,0)
s
(0,0)
(1,0)
(0,1)
(0,1)
(0,0)
(1,0)
$\nabla \mathbf{x} = \sum_{v \in s} \mathbf{J}_v . \mathbf{x}^i_v$
$\mathrm{FD} = \begin{pmatrix} x_{10} - x_{00} \\ x_{01} - x_{00} \end{pmatrix}$
$\nabla \mathbf{x} = \mathbf{x}_{01} \begin{pmatrix} 0 \\ 1 \end{pmatrix} + \mathbf{x}_{10} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \mathbf{x}_{00} \begin{pmatrix} -1 \\ -1 \end{pmatrix}$

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

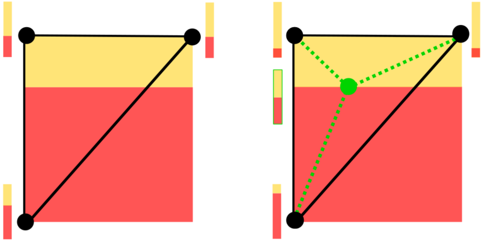

Figure 6: Updated data term after adding a new vertex.

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

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

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

[9]. 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:

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

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 $\lambda_f^i=0$, $\forall f\in\partial\Omega$. Combining (21) and (22), we get the (metric) energy for the Raviart-Thomas discretisation:

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

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.

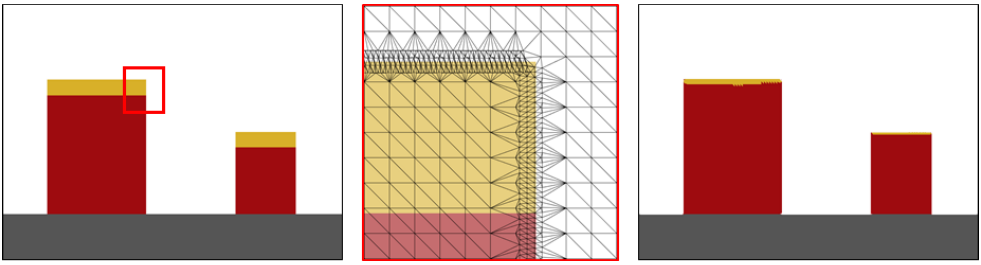

Figure 7: *Left:* Synthetic 2D scene. Colors indicate ground (*gray*), building (*red*) and roof (*yellow*). *Middle:* Zoom of the *control mesh*. *Right:* Reconstructed semantic 2D model.

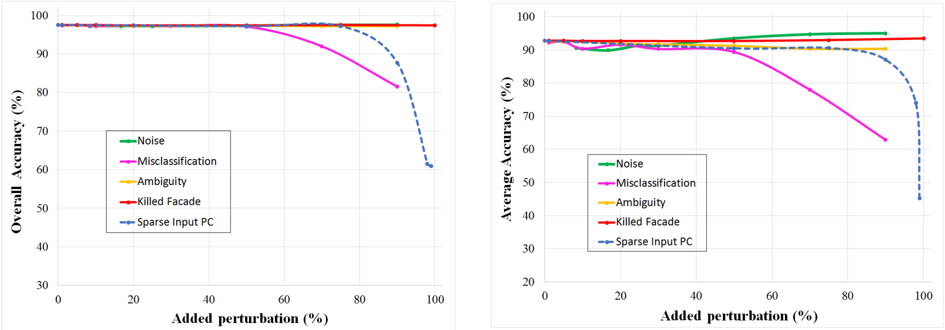

Figure 8: Quantitative evaluation of Raviart-Thomas FEM method *w.r.t.* different degradations of the input data.

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

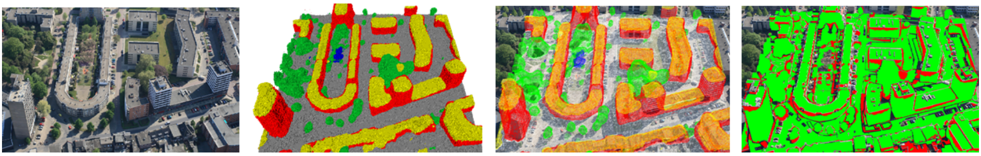

Figure 9: Quantitative evaluation of Scene 1 from Enschede. *Left:* One of the input images. *Middle left:* Semantic 3D model obtained with our Raviart-Thomas FEM method. *Middle Right:* Back-projected labels overlayed on the image. *Right:* Error map, misclassified pixels are marked in red.

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].

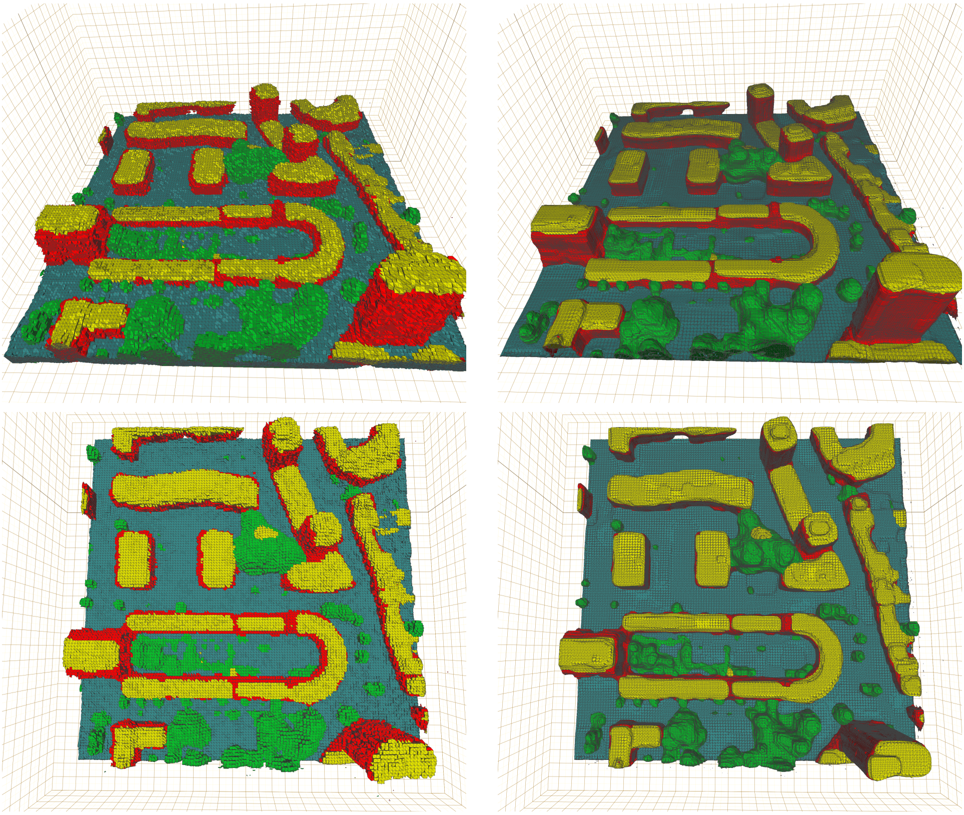

Figure 10: Reconstruction with Raviart-Thomas (*left*) and with the Lagrange basis (*right*). We deliberately select a low resolution, choose flat shading and plot mesh edges, to accentuate the differences. Please refer to the text for details.

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 multiresolution 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.