3D All The Way: Semantic Segmentation of Urban Scenes From Start to End in 3D

Anđelo Martinović[♠1] Jan Knopp[♠1] Hayko Riemenschneider[2] Luc Van Gool[1,2]
[1]KU Leuven [2]ETH Zurich

Abstract

We propose a new approach for semantic segmentation of 3D city models. Starting from an SfM reconstruction of a street-side scene, we perform classification and facade splitting purely in 3D, obviating the need for slow image-based semantic segmentation methods. We show that a properly trained pure-3D approach produces high quality labelings, with significant speed benefits (20x faster) allowing us to analyze entire streets in a matter of minutes. Additionally, if speed is not of the essence, the 3D labeling can be combined with the results of a state-of-the-art 2D classifier, further boosting the performance. Further, we propose a novel facade separation based on semantic nuances between facades. Finally, inspired by the use of architectural principles for 2D facade labeling, we propose new 3D-specific principles and an efficient optimization scheme based on an integer quadratic programming formulation.

1. Introduction

Increasingly, the topics of recognition and 3D reconstruction are intermingled. On the one hand, adding 3D features may aid recognition, on the other the knowledge about object classes helps with their 3D modeling. In the end, one can imagine feedback loops - cognitive loops if you will (e.g. [47]) - where a system jointly evolves through the solution spaces that each such subproblem (e.g. recognition, 3D reconstruction) lives in. Human vision seems to strive for such kind of unified, consistent interpretation and the endeavour seems to serve us well.

This paper looks into the creation of semantic, 3D models of cities. The task comes with both the subtasks of recognition and 3D modeling. Thus, the models should not only consist of high-quality 3D models, but ought to come with delineated functional units (e.g. windows, doors, balconies, etc.). Although substantial effort has already gone into the creation of 3D city models, efforts to render those 'semantic' are rather recent. One of the most important steps to that effect is semantic facade parsing. Thus far, it has been largely treated as a 2D pre-processing step. This paper investigates whether it could benefit from a direct coupling to the 3D data that mobile mapping campaigns also produce. As a matter of fact, the paper presents an entire pipeline for facades, from raw images to semantic 3D model, with all steps carried out in 3D. As we will show, our avoidance of any going back and forth between 2D and 3D leads to substantially shorter runtimes (20x faster).

In particular, we see three main contributions:

- an end-to-end facade modelling fully in 3D,
- a novel facade separation based on the results of semantic facade analysis,
- a formulation and implementation of weak architectural principles like alignment, symmetry, etc. in 3D.

2. Related work

As this work joins multiple research areas for the purpose of 3D facade understanding and modeling, we will briefly review the current state of the art in (1) 3D classification, (2) facade parsing and (3) facade separation.

2.1. 3D classification

A vast amount of work has dealt with the issue of 2D classification. However, the bulk of 3D classification work is rather recent, especially where 2D and 3D features are used together. To the best of our knowledge, [9] were the first to combine image classification and sparse 3D points from SfM. [25] combine depth maps and appearance features for better classification. In a similar vein, [32, 2] combine LIDAR and image data.

Since then, recent works show the advantage of combining 3D and 2D for classification [42, 21, 19, 38, 56] or place recognition [37] using LIDAR, 3D CAD or Kinect models. However, 3D is used in the context of improving 2D recognition, as these approaches still heavily rely on 2D features. Some even show that 2D information is much more important than 3D descriptors [38]. In contrast, instead of using 3D as useful aid for the 2D classifier, we design an

♠ Indicates equal contribution.

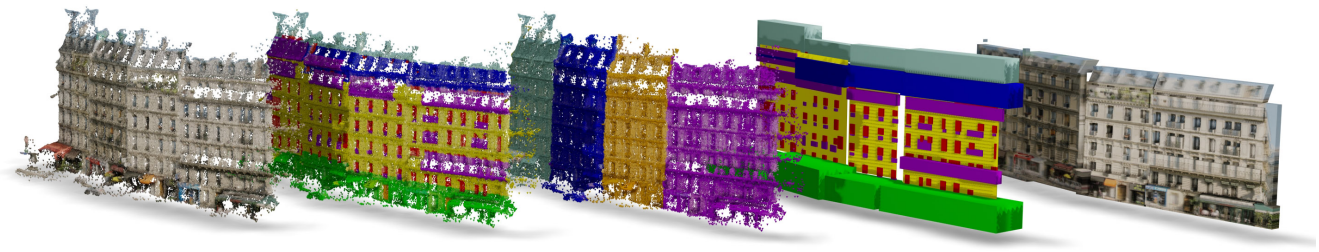

Figure 1. **Our approach.** In contrast to the majority of current facade labeling methods, our approach operates completely in 3D space. From left to right: (a) image-based SfM 3D point cloud (b) initial point cloud classification (c) facade splitting (d) structure modeling through architectural principles and (e) projected original images onto estimated 3D model. The advantages of pure-3D approach range from tremendous speed-up to complementarity with 2D classifiers.

exclusively 3D pipeline as a fast alternative to previous approaches, with competitive results.

In the 3D-only domain, a variety of local descriptors have been introduced in recent years. Unfortunately, the best performing features are typically expensive to calculate, or limited to e.g. manifold meshes [20, 8, 22]. Furthermore, automatically obtained 3D is incomplete, containing noise, holes and clutter. Thus, spin images [18] are still a popular choice (shown to be robust in the presence of noise [36]), combined with several low-level features such as color, normals, histograms etc. [15, 29, 42, 33, 34, 43]. We follow this vein of research, carrying out facade labeling completely in 3D: from using simple 3D features, point-based classification with Random Forests, and with a 3D Conditional Random Field smoothing. This results in competitive results with significant speed benefits.

2.2. Facade parsing

For street scenes, classical image segmentation techniques [40] have been extended with architectural scene segmentation using color and contour features [3]. Additional sources of information such as a height prior [53, 54, 9] or object detectors [35, 27] are typically introduced on top of local features. However, classification is performed mainly on 2D images, whereas 3D is introduced only at a procedural level [46, 41, 30].

To capture the structure inherent to facades, different approaches have been proposed. Several utilize shape grammars to learn and exploit the structure in facades.

[1] model facades with stochastic context-free grammars and rjMCMC sampling. [31] use regular grids to infer a procedural CGA grammar for repetitive facades. [39] assume multiple interlaced grids and provide a hierarchical decomposition. A similar assumption of block-wise decompositions can be used to parse facades using a binary split grammar [55]. [45] use a specialized Haussmannian facade grammar coupled with a random walk optimization. [44] extend this with a Reinforcement Learning-based optimization. [41] additionally use 3D depth information in a GA optimization framework. [11] propose efficient DP subproblems which hard-code the structural constraints. Moving further away from hard-coded shape grammars, [35] use irregular lattices to reduce the dimensionality of the parsing problem, modeling symmetries and repetitions. [24] relaxes the Haussmannian grammar to a graph grammar where structure and position are optimized separately.

Moving entirely away from strict grammars, [12] use a facade-specific segmentation together with learning weights for different meta-features capturing the structure. [48] model alignment and repetition through spatial relations in a CRF framework. [27] suggest a three-layered approach introducing 2D weak architectural principles instead of rigid shape grammar rules.

To impose as few restrictions on the facade structure as possible, we build upon the latter work by proposing novel weak 3D architectural principles and an elegant optimization formulation based on integer programming. This allows us to handle a larger variety of architectural styles and to obtain better facade reconstructions with significant speedups over their 2D counterparts.

2.3. Facade separation

All of the aforementioned works on facade parsing assume individual facades to be separated beforehand. Yet, this separation is not trivial by far and quite a few automated pipelines gloss over the issue.

In the full 3D domain, most work focuses on building extraction, which deals with 2.5D height models and identifies individual building blocks and their roof types based on height information [26].

Similar approaches have been adopted for street-level data, where height is rightfully used as the most discriminative feature [54, 58]. Other methods deal with repetitive features that reoccur on one facade and not on neighboring facades [50]. These work well if the assumptions are correct, i.e. the buildings have different heights and a quite different appearance of their facades. However, some architectural styles aim at similar appearances and heights. Furthermore, methods working in 2D require the facades to be completely visible in each image.

As an alternative to these approaches, we propose a semantic facade separation which exploits the usually varying layout of semantic structures in different facades. In contrast to methods for facade structure understanding [28, 49] which require already split facades, we propose to use the semantic scene knowledge to create these splits.

3. Our approach

Our goal is to estimate a semantically segmented 3D scene starting from images of an urban environment as input. As a first step, we obtain a set of semi-dense 3D points from standard SfM/MVS algorithms [52, 14, 17].

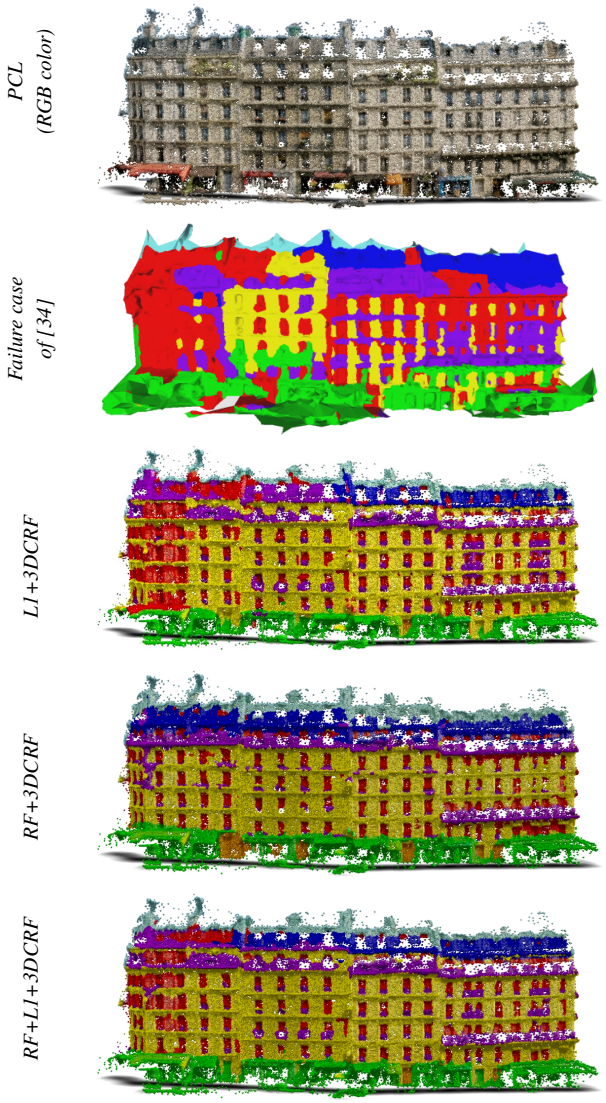

Figure 2. **Qualitative results.** Extremely challenging subset of RueMonge2014, dubbed Sub28 in [34]. Interestingly, the 2D and 3D-based methods (third vs. fourth row) outperform each other for different parts of the scene, while their combination (fifth row) has the best performance.

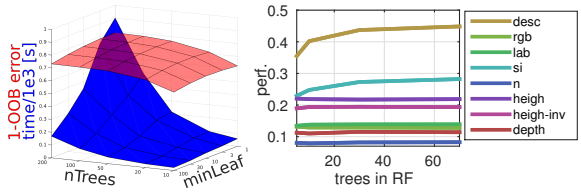

Figure 3. **Parameters of the 3D classifier.** Left: performance (red) and test time (blue) w.r.t. the number of trees and the minimum number of observations per leaf. Right: the performance of each descriptor part individually (and the final descriptor-desc) using RF classifier.

Next, we classify each point P_i in the point cloud into one semantic class L_i (window, wall, balcony, door, roof, sky, shop), using a Random Forest classifier trained on light-weight 3D features (Sec. 3.1). Afterwards, we separate individual facades by detecting differences in their semantic structure (Sec. 3.2). Finally, we propose architectural rules that express preferences such as the alignment or co-occurrence of facade elements. These rules have two effects: they improve our results and directly return the high-level 3D facade structure (Sec. 3.3).

3.1. Facade labeling

We create the initial labeling of the 3D scene by employing a Random Forest (RF) classifier on the following set of descriptors for each 3D point P_i: mean **RGB** colors of the point as seen in the camera images; the **LAB** values of that mean RGB [34]; normal (**n**) at the 3D point; 3D geometry captured using the spin-image (**SI**) descriptor [18], calculated on different scales; the point's height (**h**) above the estimated ground plane; its "inverse height" ($\mathbf{h}^{-1}$), defined as the distance from the uppermost point of the facade in the direction of the gravity vector; depth (**dph**) defined as the distance of the point P_i to the approximate facade plane. Since we do not have the facade separation available yet, we estimate $\mathbf{h}^{-1}$ and **dph** from the subset of 3D points assigned to their nearest camera. Thus, the full descriptor per point P_i is:

$$\underset{132\times1}{\mathbf{d}_i} = [\underset{3\times1}{\mathbf{RGB}_i^\mathsf{T}}\ \underset{3\times1}{\mathbf{LAB}_i^\mathsf{T}}\ \underset{3\times1}{\mathbf{n}_i^\mathsf{T}}\ \underset{120\times1}{\mathbf{SI}_i^\mathsf{T}}\ \underset{1\times1}{\mathbf{h}_i^\mathsf{T}}\ \underset{1\times1}{\mathbf{h^{-1}}_i^\mathsf{T}}\ \underset{1\times1}{\mathbf{dph}_i^\mathsf{T}}]^\mathsf{T}.$$

Once this 132-dimensional descriptor is known for each point P_i, we train an RF classifier with a uniform class prior. All classification parameters, such as scales of the SI descriptor $(0.15, 0.3, 0.45)$, number of trees (100) and minimum leaf size (30) in the RF are determined using grid search on out-of-bag (OOB) error estimates. The effect of these parameters and the impact of each utilized 3D descriptor on classifier performance are shown in Figure 3.

3.2. Facade splitting

Given the point cloud $P = \{P_i\}$ and its labeling results $L = \{L_i\}$ with the best class label L_i assigned to each individual 3D point P_i, we propose a novel method for separating individual facades. The underlying issue with previous work is that typical features such as height or appearance are too weak, especially in strongly regulated urban scenes, such as Haussmannian architecture in Paris.

We propose a facade separation method that exploits semantic nuances between facades. Despite the strong similarity of buildings, even in Haussmannian style, each facade shows individual characteristics such as window heights, balcony placements and roof lines. This knowledge is only available after performing semantic classification.

In order to separate facades into individual units, we vertically split the dominant facade plane, by posing a labeling problem that assigns sites $S = \{s_i\}$ (single connected components within the classified point cloud P) to facade groups $G = \{g_i\}$ is defined as:

$$E(S) = \sum \Theta(g_i, s_i) + \lambda \cdot \sum \Psi(s_i, s_j) \qquad (1)$$

where $\Theta(g_i, s_i)$ determines the cost of assigning a site s_i to a facade group g_i, equal to its distance in 1D location. The pairwise term $\Psi(s_i, s_j)$ encourages splits where there is decreased likelihood of crossing any architectural elements, such as windows or balconies. It aggregates the class labels in vertical direction and estimates a ratio between wall class (where the split should occur) and structural classes (such as windows, balconies, etc. where no split should be selected).

Each facade group g_i is a label which defines a candidate layout for an individual facade. It is determined by clustering features F capturing the differences between semantic elements. These features are statistical measurements defined as

$$\mathbf{F_i} = [\delta_i^\mathsf{T},\ \mathbf{A}_i^\mathsf{T},\ \mathbf{Major}_i^\mathsf{T},\ \mathbf{Minor}_i^\mathsf{T},\ \mathbf{verthist}(C_i)] \quad (2)$$

where for each connected component, δ is the position along the dominant plane, **A** is its area, **Major**, **Minor** are the lengths of its axes and **verthist** is the histogram over the class labels above and below this connected component. These features are clustered using Affinity Propagation [13] and determine the facade groups G.

The final assignment is optimized with a multi-label graphcut [7, 6, 23], which assigns all original 3D points P w.r.t. their distance to one of the facade groups G, and the final labeling determines the number of facades.

For intermediate evaluation, we manually annotate the splits between individual facades, and evaluate how many facades were correctly separated. The achieved accuracy in terms of correct facade-wise classification is 95.5% where all but 3 facades have at least 97% 3D points assigned correctly. A baseline using F=RGB+intensity gives only 78% overall, failing to split four and oversplitting additional five.

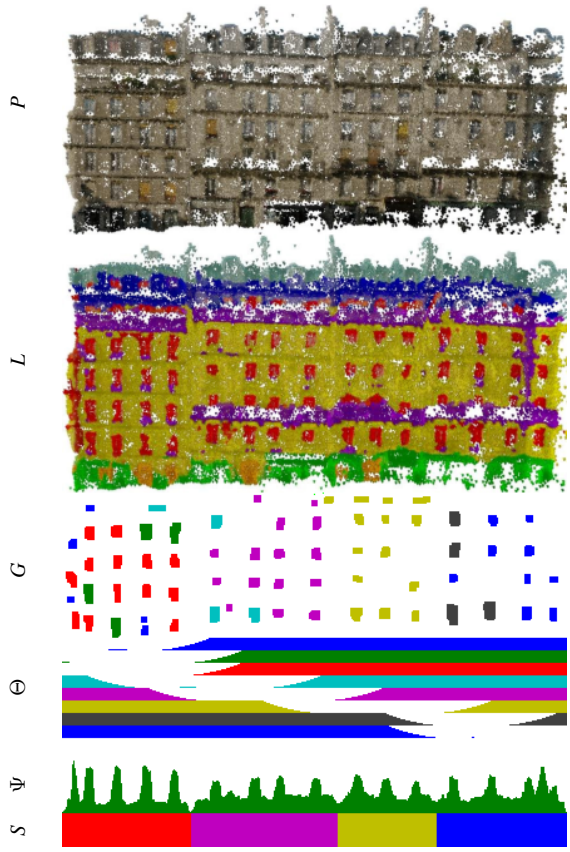

Figure 4. Exemplar facade split projected into 2D for visualization (top to bottom): 3D colored points, 3D classification, group prototypes (here windows), unary/pairwise costs and final 1D group assignment. Note the high similarity in appearance and height.

3.3. Weak architectural rules (WR)

In order to generate a more structured final output, and inspired by the state-of-the-art 2D facade labeling approach [27] we use generic rules such as *alignment*, *symmetry*, *co-occurrence*, and *vertical region order*. In contrast to shape grammars [44], these rules are more generic, and can be used for different architectural styles. The main idea of this approach is that some rules are used to discover the candidate objects in the facade, while others score the elements or modify their position and size.

3.3.1 3D rules (3DWR)

We propose a generalization of the 3D architectural principles to 3D with several major differences. Unlike [27], where initial elements are determined with a simple connected component analysis on a binary 2D image, we discover them in a robust way directly in the point cloud. Sec-

ond, since our approach works with 3D boxes instead of bounding rectangles, our approach implicitly models the z-position (along the facade normal) and depth of each facade element. Furthermore, we generalize the alignment rule to synchronize same-class elements in the z-direction. This allows us to naturally model facade configurations with inset windows or extruding balconies. Finally, we formulate our 3D principles in a novel, elegant optimization framework.

Our goal is to find the optimal set of boxes ($\mathcal{B}$) which (1) fits well to the initial data labeling (L); (2) has well *aligned* boxes; (3) does not contain overlapping elements of the same class; (4) satisfies element *co-occurrence*, e.g. a balcony should not appear if there is no window above it. We formulate this optimization as follows:

$$\begin{aligned} \underset{\mathcal{B}\in\mathcal{B}^{\text{super}}}{\arg\min}\big(f_{\text{data}}(\mathcal{B},P,L)+f_{\text{align}}(\mathcal{B})\big). \\ s.t. \quad c_{\text{overlap}}(\mathcal{B})=0 \\ c_{\text{co-occ}}(\mathcal{B})=0 \end{aligned} \tag{3}$$

Generating the initial set of boxes. From our initial point cloud labeling L, we generate an over-complete set of boxes $\mathcal{B}^{\text{super}}$. Note that in [27], the initial elements are generated by finding connected components in a labeled 2D image, followed by fitting of minimal bounding rectangles. Performing the similar task in 3D raises two main issues. First, we cannot use the 4- or 8-connected neighborhood to discover the connected components, as we deal with 3D points in continuous space. Second, the 2D approach often generates too large elements, e.g. in presence of significant noise, when distinct facade elements appear connected in L.

In our approach, for each object class c (window, balcony, door) we create a binary labeling L^c, where $L^c_i = 1$ if $L_i = c$ and 0 otherwise. We extract the initial facade elements from the labeling L^c by creating a K-nearest neighbor graph in 3D ($K = 7$ in our experiments), and discarding edges that connect nodes labeled with 1 and 0. We fit a 3D box to each component, and add it to $\mathcal{B}^{\text{super}}$.

However, since the labeling L^c can be quite noisy, we clean it up with the generalization of the morphological *opening* (erosion followed by dilation) operator to 3D. The erosion operator changes the label of a point to 0 if any of its K nearest neighbors is labeled with 0, while the dilation performs the opposite process. By varying the number of subsequent erosions and dilations, we generate multiple overlapping proposals for each facade element, with different degrees of smoothing – all used to augment $\mathcal{B}^{\text{super}}$.

Finally, we use the *symmetry* principle to add elements which are potentially missing in the scene. We detect the main vertical symmetry plane of the facade, mirror all elements and add them to $\mathcal{B}^{\text{super}}$.

Best set of boxes. We pose the search problem in Eq. 3 as an integer quadratic program (IQP) with linear constraints. Each box $\mathcal{B}_i \in \mathcal{B}$ is assigned an indicator variable $\mathbf{x_i} \in \mathbf{x}$, which is equal to 1 if the box is selected in the set, 0 otherwise. The IQP is formulated as follows:

$$\begin{aligned} \min \quad & \mathbf{w}^T_{\text{data}}\mathbf{x}+\frac{1}{2}\mathbf{x}^T\mathbf{Q}_{\text{align}}\mathbf{x} \\ s.t. \quad & \mathbf{C}_{\text{overlap}}\mathbf{x}\le\mathbf{1} \\ & \mathbf{C}_{\text{co-occ}}\mathbf{x}\ge\mathbf{0} \\ & \mathbf{x}_i\in\{0,1\} \end{aligned} \tag{4}$$

For each box B_i with label c, we set the *data* term $\mathbf{w}_{\text{data}}(i) = |L(\mathcal{B}_i) = c| - |L(\mathcal{B}_i) \neq c|$, and then normalize $\mathbf{w}_{\text{data}}$ to sum up to unity.

The *alignment* term is defined for pairs of boxes $\mathcal{B}_i$ and $\mathcal{B}_j$. We distinguish 6 types of alignment: top and bottom, left and right, back and front. For each type, two boxes are aligned if the corresponding edges of the boxes are within a threshold (equal to half the median size of objects of the same class, in the appropriate direction).

$$\mathbf{Q}_{\text{align}}(i,j)=\begin{cases}-a & \text{if } \mathcal{B}_i \text{ and } \mathcal{B}_j \text{ are aligned } a \text{ times}\\ 0 & \text{otherwise.}\end{cases} \tag{5}$$

To make sure that the resulting quadratic program is convex, we make the resulting matrix diagonally dominant, and therefore positive semi-definite:

$$\mathbf{Q}_{\text{align}}(i,i)=\sum_{j,j\neq i}|\mathbf{Q}_{\text{align}}(i,j)| \tag{6}$$

Every row of the *overlap* constraint matrix $\mathbf{C}_{\text{overlap}}$ ensures that a pair of same-class overlapping boxes (IOU>0) $\mathcal{B}_i$ and $\mathcal{B}_j$ cannot be selected at the same time ($\mathbf{x}_i + \mathbf{x}_j \le 1$). The *co-occurrence* principle prohibits balconies without at least one window on top:

$$\mathbf{C}_{\text{co-occ}}(i,j)=\begin{cases}-1 & \text{if } i=j \text{ and } \mathcal{B}_i \text{ is a balcony.}\\ 1 & \text{if } \mathcal{B}_i \text{ is a balcony and}\\ & \mathcal{B}_j \text{ is an adjacent window.}\\ 0 & \text{otherwise.}\end{cases} \tag{7}$$

The optimization in Eq. 4 is solved using the MOSEK mixed-integer solver via the CVX software package [16].

Locations and sizes of boxes. The optimization of Eq. 3 does not modify the size or location of the selected boxes. Following [27], we use a modified alignment principle. The objective function is defined as the sum of Tukey's bounded influence functions [57] evaluated on absolute differences of bounding box edge positions, for all pairs of boxes. We solve for the box locations and sizes using a Quasi-Newton optimization approach. In essence, this optimization "snaps" the borders of nearly-aligned elements to common alignment lines. Note that in the 3D case, this process implicitly results in depth-aligned windows and balconies.

Point cloud labeling		Low-res PCL		High-res PCL	
Method		Accuracy	Timing	Accuracy	Timing
3D	RF+MAP	51.42	15min	55.65	76min
	RF+3D CRF	**52.09**		**56.39**	
2D	L1+majority vote	54.68	302min	53.37	305min
	L1+MAP	55.35		54.06	
	L1+3D CRF	55.72		54.30	
	L2+majority vote	56.10	382min	54.74	385min
	L2+MAP	55.95		54.71	
	L2+3D CRF	**56.32**		**54.95**	
3D+2D	[34]	42.32	15min	39.92	23min
	RF+L1+MAP	60.16	317min	61.15	381min
	RF+L1+3D CRF	60.05		61.21	
	RF+L2+MAP	**60.44**	397min	61.31	461min
	RF+L2+3D CRF	60.43		**61.39**	

Table 1. Semantic segmentation of point clouds: accuracy for various methods on the RueMonge2014 dataset. We report the results on the low- and high-resolution point clouds as PASCAL IOU accuracy in %. The evaluation time includes feature extraction, classification, and optional 3D projection.

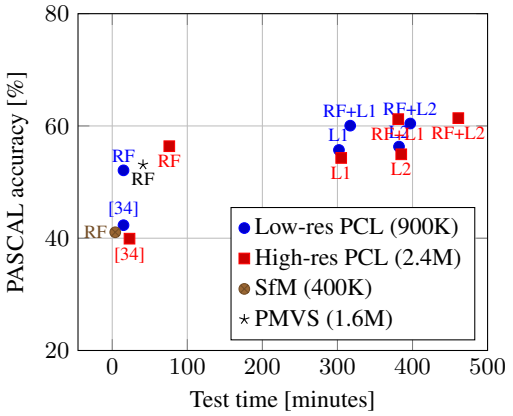

Figure 5. PCL labeling: accuracy vs. test time for two different point cloud resolutions generated by CMP. We also show the performance of our RF method on the point clouds generated by SfM and PMVS. The number of vertices per cloud is shown in parentheses.

3.3.2 Ortho + 2D rules (2DWR)

The existing 2D method [27] requires rectified 2D images and labelings as input to its third layer. Therefore, we create a "virtual 2D" input for each facade. We start by a least-square plane fit to the 3D points of the facade. The points P and their labels L are then projected onto the plane. The ortho labeling is generated by uniformly sampling points on this plane, and finding the nearest projected point for each pixel. The downside of this method is that useful 3D information is lost in the process. Furthermore, the processing time is increased due to the overhead of 2D-3D projections.

4. Evaluation

We consider three tasks pertaining to facade labeling: *point cloud labeling*, *image labeling*, and *facade parsing*. In all experiments, to evaluate the semantic segmentation results, we use the PASCAL-VOC IoU segmentation accuracy, averaged per class and shown qualitatively in Figure 8.

Datasets. We perform exhaustive evaluations on the only publicly available, street-side facade dataset RueMonge2014[2] introduced by [34]. As we focus on point cloud labeling, we consider only the vertices from their mesh, which we name 'Low-res', since it was generated by 2.7x subsampling the original, 'High-res' mesh produced by the CMPMVS algorithm [17] in 270 minutes. For reconstruction speedup, one could use [4, 5] who densely reconstruct the scene on a single core in roughly two seconds/image (14 min), or the commercial version of the CMPMVS algorithm [10] which reconstructs the same scene on the GPU in only 4 minutes. For completeness, we evaluate our 3D classifier on two additional point clouds: sparse 'SfM' (using VisualSFM [51], 13 min) and semi-dense 'PMVS' (using [14], 21 min).

[2]http://varcity.eu/3dchallenge/

4.1. Point cloud labeling

We compare several approaches for 3D point cloud labeling, see Table 1 and Figure 2. First, as the example of a purely-3D approach, we use our initial Random Forest classifier (RF+MAP). The result is then smoothed with a 3D Conditional Random Field (RF+3D CRF). The Potts model-based pairwise potentials are defined over a 4-nearest neighbor graph of the point cloud.

This result is compared with state-of-the-art 2D facade labeling. We use the publicly available 3-layered approach [27], and refer to the first two layers of this method as L1 (superpixel classification) and L2 (L1+object detectors+pixel CRF). The resulting semantic segmentation of images is projected and aggregated in the point cloud by either majority voting from different cameras, or using the aforementioned 3D CRF.

Finally, we combine the results of 3D and 2D methods using the CRF, resulting in a higher performance at the cost of evaluation time. We compare these hybrid methods to the recent approach that combines 2D and 3D for facade labeling [34], and observe significant improvement in quality. It is worth noting that the joint 2D+3D approach gives the best performance but at a 26x slower speed and with a modest 8% accuracy gain over the 3D-only approach. The high-res point cloud increases the performance of the 3D-only classifier by 4% but at the cost of a 5x slower speed.

4.2. Image parsing

Comparison to 2D methods is additionally performed in the image domain, by back-projecting our point cloud label-

Image labeling		Low-res PCL		High-res PCL	
Method		Accuracy	Timing	Accuracy	Timing
3D	RF+MAP	52.85	21min	57.82	85min
	RF+3D CRF	**53.22**		**58.13**	
2D	L1	54.46	299min	54.46	299min
	L2	**57.53**	379min	**57.53**	379min
3D+2D	[34]	41.34	15min	n/a	n/a
	RF+L1+MAP	61.58	324min	63.08	390min
	RF+L1+3D CRF	61.27		62.87	
	RF+L2+MAP	**61.95**	404min	**63.32**	470min
	RF+L2+3D CRF	61.73		63.13	

Table 2. Semantic segmentation of street-side images: accuracy for various methods on the RueMonge2014 dataset. The results are shown for the test set of 202 test images. The 2D results are obtained by running the first two layers (L1 and L2) of the 3Layer method [27], and projecting the point cloud classification onto the original images. The PASCAL IOU accuracy is shown in % over the image domain.

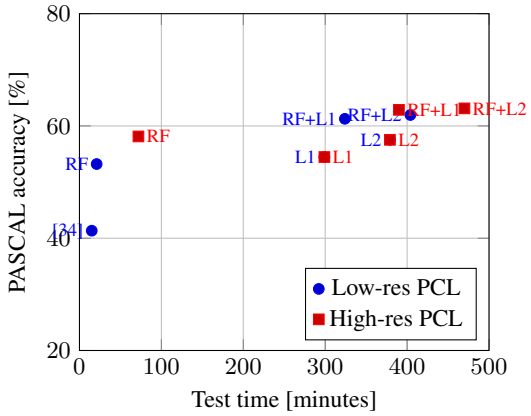

Figure 6. Image labeling: accuracy vs. test time for two different PCL resolutions.

Figure 7. **Estimated 3D facades.** All reconstructed facades in the RueMonge2014 test set. Our method performs automatic separation of facades and analyzes the 3D structure of facade elements. The final results are obtained by fitting 3D boxes to the discovered objects and texturing with ortho-images. Please zoom in to view 3D structure, or consult the detailed view in Figure 8.

ing L onto the perspective images, and filling out gaps with nearest-neighbor interpolation, see Table 2. In the image domain, a similar behavior is observed, as the 3D-only approach achieves the highest speed and competitive results to the 2D-only classification, which is only 4% better but 18x slower. The complementary combination of 2D and 3D again achieves top performance (63.32%), outperforming the existing method [34] by over 20%.

4.3. Facade parsing

We compare our 3D version of the weak architectural rules (3DWR) with its 2D counterpart (2DWR) from [27], see Table 3. The evaluation is performed in the original point cloud, by concatenating the individual facade labelings (3D) or back-projecting the labeled ortho-images (2D).

We test three different classifiers as input to this stage, based on features from 3D, 2D and 2D+3D. Our 3DWR approach outperforms its 2D counterpart in all cases except when using the 2D-only input. However, the most obvious improvement is the speed of our IQP optimization compared to the approach in [27].

Overall, top performance is achieved by a combination of 2D and 3D features and pure 3D weak architectural principles in 325 minutes (317 for initial labeling + 8 for weak rules). The fastest, yet still competitive performance uses only 3D features and 3D weak principles, which requires roughly 20 minutes from start (point cloud) to end (textured 3D models) for the full street.

A visual comparison of the stages is shown in Figure 8, including the original color point cloud, initial classification, the result of 3D weak architectural rules using the best classifier, and final geometry-correct textured models. For an overview of all facades reconstructed in 3D, see Figure 7.

5. Conclusion

In this work we proposed a new approach for 3D city modelling using 3D semantic classification, 3D facade splitting, and 3D weak architectural principles. Our method produces state-of-the-art results in terms of both accuracy and computation time. The results indicate that 3D-only classification is feasible and leads to tremendous speedups.

In future work, we plan to provide feedback to the original SfM point cloud creation to also tie in that process.

Facade parsing		Low-res PCL		High-res PCL	
Method	Input classification	Accuracy	Timing	Accuracy	Timing
2DWR [27]	3D: RF+3D CRF	49.59	802min	49.54	885min
	2D: L1+3D CRF	54.04		53.29	
	3D+2D: RF+L1+3D CRF	**58.81**		**58.40**	
3DWR (Ours)	3D: RF+3D CRF	52.24	8min	56.35	10min
	2D: L1+3D CRF	55.39		53.56	
	3D+2D: RF+L1+3D CRF	**60.83**		**59.89**	

Table 3. Semantic segmentation of street-side point clouds using weak architectural rules (WR) on the RueMonge2014 dataset. We compare the original 2D version applied on virtual ortho-images, and our proposed 3D method, for the three representative classifiers from Table 1. The PASCAL IOU accuracy is shown in %. The test time does not include the time needed for the initial point cloud classification.

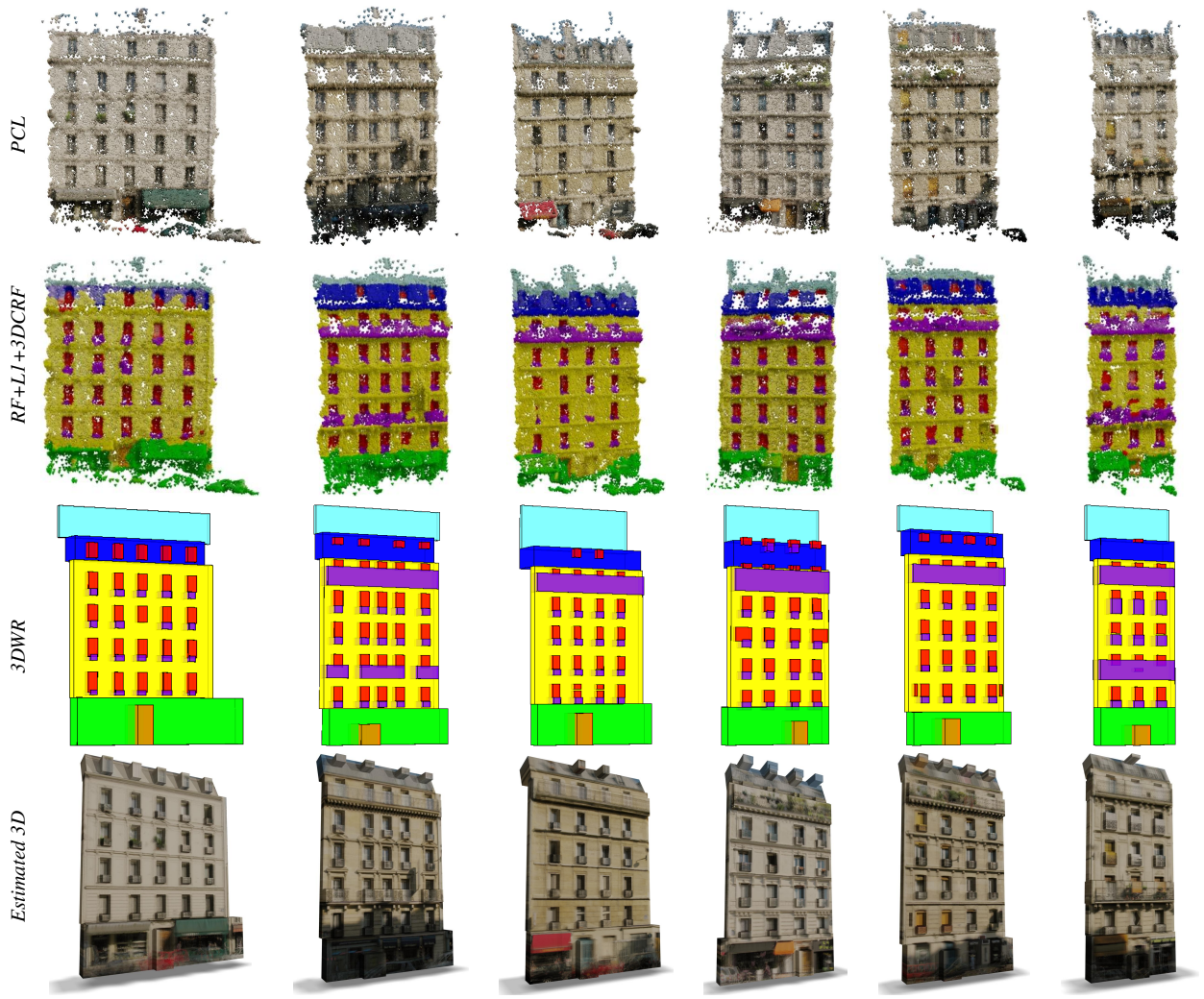

Figure 8. **Qualitative results.** We show examples of automatically obtained facades using our method. From top to bottom: initial colored point cloud (Low-res), initial classification, estimated boxes using weak 3D rules; and –as we suggested that 3D semantic interpretation can be used to estimate 3D shape– automatically generated 3D models of the facades textured by projecting ortho images.

Acknowledgements

This work was supported by the European Research Council (ERC) projects VarCity (#273940) and COGNIMUND (#240530), and by the KU Leuven Research Fund, iMinds, and FWO-project G.0861.12.

References

[1] F. Alegre and F. Dellaert. A probabilistic approach to the semantic interpretation of building facades. In *Workshop on Vision Techniques Applied to the Rehabilitation of City Centres*, 2004.

[2] D. Anguelov, B. Taskar, V. Chatalbashev, D. Koller, D. Gupta, G. Heitz, and A. Ng. Discriminative learning of markov random fields for segmentation of 3d scan data. In *CVPR*, 2005.

[3] A. Berg, F. Grabler, and J. Malik. Parsing images of architectural scenes. In *ICCV*, 2007.

[4] A. Bodis-Szomoru, H. Riemenschneider, and L. Van Gool. Fast, Approximate Piecewise-Planar Modeling Based on Sparse Structure-from-Motion and Superpixels. In *CVPR*, 2014.

[5] A. Bodis-Szomoru, H. Riemenschneider, and L. Van Gool. Superpixel Meshes for Fast Edge-Preserving Surface Reconstruction. In *CVPR*, 2015.

[6] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *PAMI*, 26(9):124–1137, 2004.

[7] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11):1222–1239, 2001.

[8] A. M. Bronstein, M. M. Bronstein, L. J. Guibas, and M. Ovsjanikov. Shape google: Geometric words and expressions for invariant shape retrieval. *ACM Trans. Graph.*, 2011.

[9] G. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV*, 2008.

[10] Capturing Reality s.r.o. RealityCapture. `http://www.capturingreality.com/`, 2014.

[11] A. Cohen, A. Schwing, and M. Pollefeys. Efficient structured parsing of facades using dynamic programming. In *CVPR*, 2014.

[12] D. Dai, M. Prasad, G. Schmitt, and L. Van Gool. Learning domain knowledge for facade labelling. In *ECCV*, 2012.

[13] B. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315:972–976, 2007.

[14] Y. Furukawa and J. Ponce. Accurate, Dense, and Robust Multi-View Stereopsis. *PAMI*, 32(8):1362–1376, 2010.

[15] A. Golovinskiy, V. Kim, and T. Funkhouser. Shape-based recognition of 3d point clouds in urban environments. In *ICCV*, 2009.

[16] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming. `http://cvxr.com/cvx`, Mar. 2014.

[17] M. Jancosek and T. Pajdla. Multi-View Reconstruction Preserving Weakly-Supported Surfaces. In *CVPR*, 2011.

[18] A. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *PAMI*, 21(5):433 – 449, 1999.

[19] O. Kaehler and I. Reid. Efficient 3D Scene Labeling Using Fields of Trees. In *ICCV*, 2013.

[20] E. Kalogerakis, A. Hertzmann, and K. Singh. Learning 3D Mesh Segmentation and Labeling. *ACM Transactions on Graphics*, 29(3), 2010.

[21] B. Kim, P. Kohli, and S. Savarese. 3D Scene Understanding by Voxel-CRF. In *ICCV*, 2013.

[22] J. Knopp, M. Prasad, and L. Van Gool. Scene cut: Class-specific object detection and segmentation in 3d scenes. In *3DIMPVT*, 2011.

[23] V. Kolmogorov and R.Zabih. What energy functions can be minimized via graph cuts? *PAMI*, 26(2):147–159, 2004.

[24] M. Kozinski and R. Marlet. Image Parsing with Graph Grammars and Markov Random Fields Applied to Facade Analysis. In *WACV*, 2014.

[25] L. Ladicky, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. Clocksin, and P. Torr. Joint Optimization for Object Class Segmentation and Dense Stereo Reconstruction. *IJCV*, 100(2):122–133, 2012.

[26] F. Lafarge, X. Descombes, J. Zerubia, and M. Pierrot Deseilligny. Structural approach for building reconstruction from a single DSM. *PAMI*, 32(1):135–147, 2010.

[27] A. Martinović, M. Mathias, J. Weissenberg, and L. Van Gool. A Three-Layered Approach to Facade Parsing . In *ECCV*, 2012.

[28] A. Martinović and L. Van Gool. Bayesian grammar learning for inverse procedural modeling. In *CVPR*, 2013.

[29] O. Mattausch, D. Panozzo, C. Mura, O. Sorkine-Hornung, and R. Pajarola. Object detection and classification from large-scale cluttered indoor scans. *EUROGraphics*, 2014.

[30] P. Müller, P. Wonka, S. Haegler, A. Ulmer, and L. Van Gool. Procedural modeling of buildings. In *SIGGRAPH*, 2006.

[31] P. Müller, G. Zeng, P. Wonka, and L. Van Gool. Image-based procedural modeling of facades. In *SIGGRAPH*, 2007.

[32] D. Munoz, J. Bagnell, and M. Hebert. Co-inference for Multi-modal Scene Analysis. In *ECCV*, 2012.

[33] L. Nan, K. Xie, and A. Sharf. A search-classify approach for cluttered indoor scene understanding. *SIGGRAPH Asia*.

[34] H. Riemenschneider, A. Bodis-Szomoru, J. Weissenberg, and L. Van Gool. Learning Where To Classify In Multi-View Semantic Segmentation. In *ECCV*, 2014.

[35] H. Riemenschneider, U. Krispel, W. Thaller, M. Donoser, S. Havemann, D. Fellner, and H. Bischof. Irregular lattices for complex shape grammar facade parsing. In *CVPR*, 2012.

[36] S. Salti, A. Petrelli, and F. Tombari. On the affinity between 3d detectors and descriptors. In *3DPVT*, 2012.

[37] T. Sattler, B. Leibe, and L. Kobbelt. Fast image-based localization using direct 2d-to-3d matching. In *ICCV*, 2011.

[38] S. Sengupta, J. Valentin, J. Warrell, A. Shahrokni, and P. Torr. Mesh Based Semantic Modelling for Indoor and Outdoor Scenes. In *CVPR*, 2013.

[39] C.-H. Shen, S.-S. Huang, H. Fu, and S.-M. Hu. Adaptive partitioning of urban facades. *ACM Trans.Graph.*, 30(6):184, 2011.

[40] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *CVPR*, 2008.

[41] L. Simon, O. Teboul, P. Koutsourakis, L. Van Gool, and N. Paragios. Parameter-free/pareto-driven procedural 3d reconstruction of buildings from ground-level sequences. In *CVPR*, 2012.

[42] S. Song and J. Xiao. Sliding shapes for 3d object detection in depth images. In *ECCV*, 2014.

[43] M. Sunkel, S. Jansen, M. Wand, and H.-P. Seidel. A Correlated Parts Model for Object Detection in Large 3D Scans. *EUROGraphics*, 2013.

[44] O. Teboul, I. Kokkinos, L. Simon, P. Koutsourakis, and N. Paragios. Parsing facades with shape grammars and reinforcement learning. *PAMI*, 35(7):1744–1756, 2013.

[45] O. Teboul, L. Simon, P. Koutsourakis, and N. Paragios. Procedural modeling and image-based 3d reconstruction of complex architectures through random walks. In *IJCV*, 2010.

[46] O. Teboul, L. Simon, P. Koutsourakis, and N. Paragios. Segmentation of building facades using procedural shape prior. In *CVPR*, 2010.

[47] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, and L. Van Gool. Depth-from-recognition: Inferring metadata by cognitive feedback. In *ICCV*, 2007.

[48] R. Tyleček and R. Šára. Spatial pattern templates for recognition of objects with regular structure. *Pattern Recognition*, 8142:364–374, 2013.

[49] J. Weissenberg, H. Riemenschneider, M. Prasad, and L. Van Gool. Is there a procedural logic to architecture? In *CVPR*, 2013.

[50] A. Wendel, M. Donoser, and H. Bischof. Unsupervised facade segmentation using repetitive patterns. In *DAGM*, 2010.

[51] C. Wu. VisualSFM: A Visual Structure from Motion System. `http://ccwu.me/vsfm/`, 2011.

[52] C. Wu. Towards linear-time incremental structure from motion. In *3DPVT*, 2013.

[53] J. Xiao, T. Fang, P. Tan, P. Zhao, E. Ofek, and L. Quan. Image-based facade modeling. *ACM Graphics*, 2008.

[54] J. Xiao and L. Quan. Multiple view semantic segmentation for street view images. In *ICCV*, 2009.

[55] C. Yang, T. Han, L. Quan, and C.-L. Tai. Parsing façade with rank-one approximation. In *CVPR*, 2012.

[56] C. Zhang, L. Wang, and R. Yang. Semantic segmentation of urban scenes using dense depth maps. In *ECCV*, 2010.

[57] Z. Zhang, Z. Zhang, P. Robotique, and P. Robotvis. Parameter estimation techniques: A tutorial with application to conic fitting. *Image and Vision Computing*, 15:59–76, 1997.

[58] P. Zhao, T. Fang, J. Xiao, H. Zhang, Q. Zhao, and L. Quan. Rectilinear parsing of architecture in urban environment. In *CVPR*, 2010.