Technical Report: Co-learning of geometry and semantics for online 3D mapping

Marcela Carvalho, Maxime Ferrera, Alexandre Boulch, Julien Moras, Bertrand Le Saux, Pauline Trouvé-Peloux

Technical Report: Co-learning of geometry and semantics for online 3D mapping

Marcela Carvalho†, Maxime Ferrera†, Alexandre Boulch, Julien Moras, Bertrand Le Saux, Pauline Trouvé-Peloux

DTIS, ONERA, Université Paris Saclay F-91123 Palaiseau - France
† Equal Contribution

Abstract. *This paper is a technical report about our submission for the ECCV 2018 3DRMS Workshop Challenge on Semantic 3D Reconstruction [1]. In this paper, we address 3D semantic reconstruction for autonomous navigation using co-learning of depth map and semantic segmentation. The core of our pipeline is a deep multi-task neural network which tightly refines depth and also produces accurate semantic segmentation maps. Its inputs are an image and a raw depth map produced from a pair of images by standard stereo vision. The resulting semantic 3D point clouds are then merged in order to create a consistent 3D mesh, in turn used to produce dense semantic 3D reconstruction maps. The performances of each step of the proposed method are evaluated on the dataset and multiple tasks of the 3DRMS Challenge, and repeatedly surpass state-of-the-art approaches.*

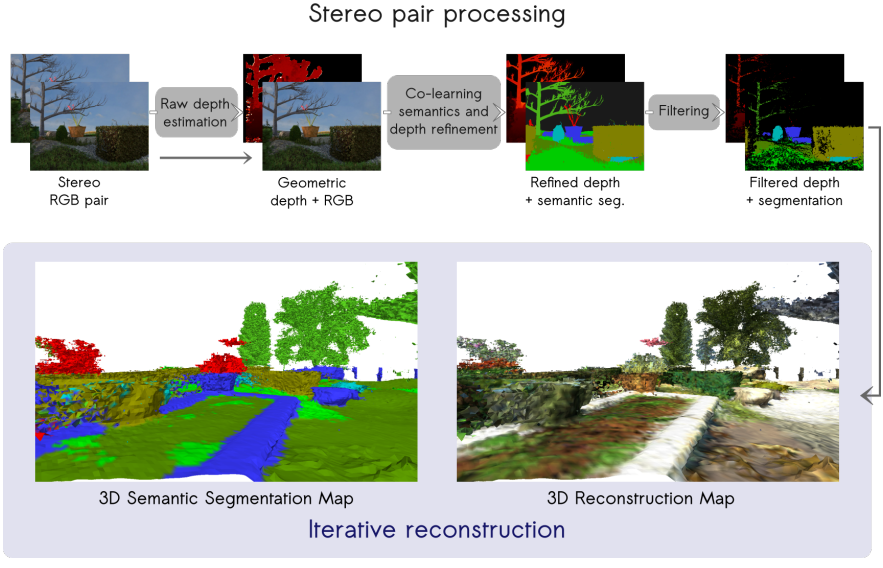


Fig. 1. Pipeline for generating geometric and semantic 3D reconstruction maps.

1 Introduction

Autonomous navigation is conditioned by the ability of sensing and analyzing the environment to take new decisions. In this context, accurate 3D reconstruction and semantic understanding of the scenes are critical. Indeed, building a 3-Dimensional (3D) map of the scene including semantic information allows to plan future trajectories accordingly to the tasks to perform.

Over the past years, improvements on data acquisition techniques and processing made possible reconstructing 3D scenes in multiple ways. On the one hand, active sensors are now mature technology and some variants gain special attention, like LIDARs, which produce dense and reliable point clouds [2]; and RGB-D sensors that generates corresponding depth maps which can be combined to scene reconstruction [3]. On the other hand, passive approaches like Structure-from-Motion (SfM) are also commonly adopted to recover 3D relations between points and objects from a set of 2D images. In SfM, we distinguish offline methods from online methods. Offline variants, also denoted as photogrammetry, usually exhaustively process all data before global reconstruction. In opposition, online approaches handle information incrementally to perform reconstruction while data are being acquired, as in Simultaneous Localization and Mapping (SLAM) and in Visual Odometry (VO). However, these latter techniques usually rely on geometric features and not on semantic information, though it is an important feature to perform more specialized and complex navigation tasks.

In this work, we present a new approach to jointly learn geometry and semantics for 3D mapping. The proposed pipeline consists of two steps, corresponding to different levels of data aggregation (Fig. 1). First, at image level, a multi-task network estimates a depth map and a semantic segmentation map. Then, these geometric and semantic features are accumulated into a global representation where the semantic mesh of the scene is extracted from the 3D representation, which allows scene understanding and planning of further actions.

In details, the main contributions of this paper are the following. The first key point is the joint use of geometric and machine learning approaches. As illustrated in Figure(1), a raw depth map is estimated from a pair of images using stereo and then is refined through a convolutional neural network. A second key point is the co-learning of depth and semantic segmentation from the raw depth map and an RGB images. Hence the proposed network performs multiple tasks at once, with mutual benefit. We show that this approach leads to better performances than independent predictions of depth and semantic segmentation. Finally, on the contrary to global, offline reconstruction methods, our approach is incremental and hence is conceptually compatible with autonomous navigation and robotics.

The paper is organized as follows: section 2 presents works related to the problem, section 3 describes our semantic reconstruction pipeline and finally section 4 evaluates our method with quantitative and qualitative results on the *3D Reconstruction Meets Semantics 2018* (3DRMS) Challenge dataset, which contains series of stereo sequences generated over a simulated garden.

2 Related work

Perception for autonomous navigation has been a great topic of interest in the last two decades. As cameras became cheap and easy to embed while still offering rich information, vision-based SLAM methods grew more and more popular [4,5]. SLAM allows a robot to localize itself with respect to the environment. Either this environment is unknown, and its 3D structure is simultaneously estimated, or the environment is already known, and a previously built map can be used [6,7]. In the latter case, such maps can be obtained by regular SLAM methods, *i.e.*. building the map of the environment and then using it for self localization. Maps can also be built offline by SfM algorithms such as Colmap [8] or OpenMVG [9] before being used for real-time localization. All these approaches for offline or online map construction, take only the geometric structure of the scene into account. However, a few works proposed to also benefit from semantic information, yielding in *semantic SLAM* [10]. Indeed, this allows to get better maps and increase the localization reliability [11,12]. Using RGB-D data, a pipeline using random forests for creating semantic maps in 2D and 3D was proposed in [13]. More recently, [14] applied joint learning with neural networks over multiple RGB-D views to generate better 2D semantic maps, but did not reconstruct corresponding 3D models. With respect to all these approaches, ours offers a functional pipeline from 2D images to 3D reconstruction with semantics. With respect to the latter ones, semantics and geometry have a better integration directly in the network.

The **joint use of geometry and semantics** has been investigated in the previous edition of the 3DRMS challenge [15]. The dual objective was 3D geometry reconstruction and semantic classification. The proposed baseline links Colmap [8] and SegNet [16] (for 2D classification, then projected in 3D). Both entries [17,18] used semantics during the reconstruction to filter out outliers, but with worst performances than the baseline. With respect to these approaches, we propose to learn the fusion of semantics and geometry through a multi-task network.

Fully Convolutional Networks (FCNs) [19,16,20] have been widely used for many tasks in computer vision. In brief, they are dense prediction methods which intend to assign information back onto the original pixels positions. **Semantic segmentation** is a common domain of application for such dense prediction networks.We focus here on the approaches which benefit from geometric information. FuseNet [21] uses two interlaced encoders and a single decoder for semantic segmentation from RGB-D data. Alternatively, in [22], the authors introduce *residual fusion* using a small network to merge the outputs of two SegNets applied to different sensor modalities. A finer (though more complex) approach, 3D graph neural network [23], consists in considering information extracted from the local 3D graph of adjacency and using it in the segmentation network. [17] proposed 3D-consistent data augmentation to incorporate the geometry directly in the training set. Among all these approaches, the one which has most in common with ours is FuseNet [21], since they share solving the fusion problem by a highly-integrated network. However, our network goes beyond

simple fusion, and address a multi-task problem, with semantic segmentation and depth adjustment.

FCNs have also been applied to other tasks such as monocular **depth prediction** [24,25,26,27]. These techniques exploit spatial correlation based on structured information (*e.g.*, linear perspective, textures) to produce reliable depth maps from 2D scenes. To cite but a few, we then focus on approaches with open-source code. Based on SegNet, Laina *et al.* [28] exploit residual connections [29] and fast up-projection blocks. In D3-Net [27], the network consists on a densely connected encoder [30] and a U-Net like decoder structure to predict refined depth estimation. With respect to these approaches, our method also uses depth from geometry as an input, and refines the 2D depth map using semantic constraints, which yields in better depths than with stereo or monocular prediction.

3 Proposed approach

As presented in figure 1, our method is composed of two computation levels: depth and semantic maps generation; 3D data accumulation for surface reconstruction. These tasks are combined sequentially and result in an accurate method for 3D scene reconstruction. From beginning to end, we use a stereo sequence to produce a semantic mesh.

Our main idea is to learn jointly the depth and the semantic segmentation in a multi-task deep neural network framework. Besides, we also benefit from geometric depth estimation methods. Indeed, raw depth map estimated from a pair of stereo images with geometric approach are used as inputs of the multi-task network. In the following, we describe in details the four sub-tasks of Fig. 1.

3.1 Depth estimation

The first step of the proposed 3D reconstruction pipeline consists in estimating depth maps from stereo views. In brief, the calibration of stereo cameras allows estimating the relative pose of the right camera with respect to the left one, as well as their distortion parameters. Using these informations, the left and right images may be undistorted and rectified in order to be aligned. Once aligned, the depth of corresponding points in both images can be estimated from the known baseline between the cameras, their focal length and the disparity between the two points.

Two different stereo matching algorithms have been tested to compute disparity maps: ELAS [31] and SGBM [32]. ELAS is a probabilistic method based on the triangulation of robust matches to create reliable support points. These support points then serve as priors for the disparity search and allows the computation of a Maximum-A-Posteriori (MAP) estimate over the remaining pixels. ELAS was tested using the LIBELAS implementation, without post-processing. Indeed, some post-processing is commonly applied in order to get better and more dense disparity maps. However, the neural networks might be more prone

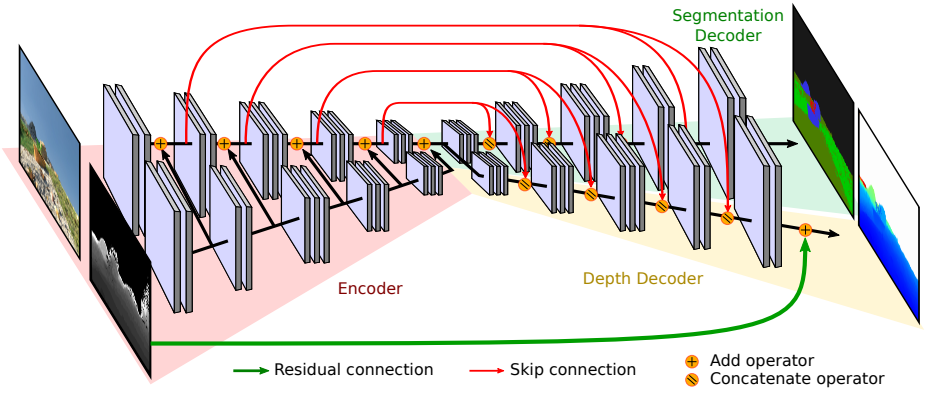


Fig. 2. Multi-task Network architecture.

to handle raw disparity maps than human-interpretable ones. Otherwise, SGBM is a semi-global method which estimates disparity by minimizing an energy function made of the Sum of Absolute Distances (SAD) over a local window and a smoothness term. SGBM was tested using its OpenCV implementation and no post-processing were applied.

Their respective accuracy results on the training sequences 0001 and 0224 of the 3DRMS dataset are displayed in table 1. SGBM gives slightly better results than ELAS, so the depth maps produced by SGBM are the ones we choose to use in the proposed pipeline.

3.2 Semantic segmentation and depth enhancement

The task at hand here is the reconstruction of a semantic mesh of the given scene. Hence, the objective is twofold: reconstruct the geometry of the scene (3D localization of the mesh vertices) and identify semantics (attach a label to each mesh element).

Though, the geometric depth estimation previously described is not sufficient for global surface reconstruction. Indeed, the small baseline of the stereo sensor might lead to huge estimation errors. Hence, a refinement step is needed to produce better depth maps. As the geometric errors mostly occur on edges, using the RGB image as an additional information would lead the network to produce sharper edges. Besides, as shown in [21,23], semantic segmentation benefits from both RGB images and depth maps.

These considerations motivate the proposed approach of a multi-task fully convolutional neural network for a joint prediction of depth and semantic segmentation. The proposed architecture was inspired in FuseNet [21] and is presented on figure 2. The Multi-task Network has an encoder-decoder structure, with two branches for the encoder, and 2 independent branches for the decoder (one for semantics, one for depth estimation). Contrarily to the original implementation of FuseNet, we add skip connections between the encoder and decoder

parts to improve spatial information flow over the network. Branches in the contractive part take the RGB and raw depth inputs respectively and as feature maps are generated, they are melt from the depth branch to the RGB input branch. Also, depth refinement is performed in a residual manner, adding the correction to the input raw map.

3.3 Filtering

Even though depth is enhanced using the multi-task network, a few errors remain when an object occludes another. In this case, the network tends to smooth the transition between objects and overlook small details (such as tree leafs for example). To avoid unwanted outliers in later stages of the 3D reconstruction, we apply the following filtering operations. First, points labeled as sky are removed. Second, points from uncertain object borders are identified and removed. These borders correspond to transitions between objects at different depths, so we compute the gradient of depth over the image and remove all pixels for which the gradient norm is greater than a fixed threshold (empirically set to 0.05). Finally, we also test in the experiments an additional filter: erosion, or removal of the neighbors of a point considered as an outlier. As we will show, this produces more precise but less complete reconstructions.

3.4 Iterative 3D map construction

The 3D reconstruction module is based on Truncated Signed Distance Function (TSDF) modeling. This technique estimates a scalar field which represents the approximate distance of every points in the 3D space to the nearest surface. In practice, the field is estimated over a 3D discretization of the world and only close to the surfaces. The distance estimates are signed: positive outside of the object and negative for the inside. Hence, the zero crossing is an implicit representation of the surfaces of the objects present in the scene and a *Marching Cube* algorithm is used to recovers the mesh. The TSDF implementation used in this paper is based on OpenChisel [33]. In order to estimate the distance field, the 3D space is discretized into voxels and the filtered depth maps are integrated into the TSDF according to the poses of the camera. The depth maps are first clipped in order to only process 3D points within a clipping range distance from the camera (in practice from 0.5 to 5-10 meters). In addition to distance estimations, we also add semantic classification fusion. Thus the module can take as a new input, either the label image resulting from classification or directly the classification scores (cf. Section 3.2). These semantic inputs are processed in the same way as the depth maps, that is the voxels integrate the semantic scores in addition to the distance-to-nearest-surface values. When all the frames have been integrated, a filter removes the voxels which do not contain accurate enough distance values. For each remaining voxel, the semantic label is selected as the one with highest score. In practice, the voxel grid resolution is set to 3cm. The mesh is finally generated by applying *Marching Cube* over this voxel grid.

Table 1. Error measurements adopted to evaluate semantic segmentation (left) and depth estimation performances (right). P corresponds to the predictions and GT the ground truth. C is the number of classes. Variables d_i and $\hat{d}_i$ are the ground truth and prediction respectively, and N is the total number of pixels.

Metric	Definition
Overall Acc. (OA)	$\frac{P \cap GT}{\lvert GT \rvert}$
Average Acc. (A.Acc.)	$\frac{1}{C} \sum_{i=1}^{C} \frac{P_i \cap GT_i}{\lvert GT_i \rvert}$
Average IoU (A. IoU)	$\frac{1}{C} \sum_{i=1}^{C} \frac{P_i \cap GT_i}{P_i \cup GT_i}$

Metric	Definition
Abs. error	$\frac{1}{N} \sum_{i=0}^{N} \frac{\lvert d_i - \hat{d}_i \rvert}{d_i}$
RMS	$\sqrt{\frac{1}{N} \sum_{i=0}^{N} (d_i - \hat{d}_i)^2}$

Table 2. Comparison of semantic segmentation results on the 3DRMS 2018 dataset using state-of-the-art segmentation networks and the proposed Multi-task Network.

Methods			Test on 0001			Test on 0224		
	Input	Output	OA	A. Prec.	A. IoU	OA	A. Prec	A. IoU
Baselines								
U-Net	RGB	S	0.9068	0.8286	0.7012	0.9054	0.7496	0.6395
FuseNet	RGB	S	0.9091	0.8577	0.7371	0.9311	0.8038	0.7169
Multitask refinement								
Multi-task Net.	RGB+D_{ELAS}	D+S	0.9363	0.8916	0.7943	0.9277	0.7952	0.7107
Multi-task Net.	RGB+D_{SGBM}	D+S	0.9411	0.8965	0.7980	0.9303	0.8017	0.7195

4 Experiments results

In this section, each step of the semantic reconstruction pipeline is evaluated on the 3DRMS dataset. The data consists in four training sequences with ground truth and a test sequence for challenge evaluation (for which the ground truth remains undisclosed to participants). We further divide the training set in train and validation to present evaluation scores and comparable visual results. Precisely, we created two folds from the training data: fold 1 with training scenes from sequences 0128, 0160, 0224 and testing scenes from sequence 0001; and fold 2 with training scenes from sequences 0001, 0128, 0160 and testing scenes from sequence 0224.

In the following, semantic segmentation (Section 4.1), depth estimation (Section 4.2) and global 3D reconstruction (Section 4.3) are evaluated on this dataset and compared to state of the art approaches. Metrics used for comparison are presented in tables 1.

4.1 Semantic segmentation

Semantic segmentation from 2D images is one of the tasks of the 3DRMS Challenge. Our architecture is evaluated against the two state-of-the-art approaches: U-Net [20] and FuseNet [21]. Performances are computed according to the metrics presented in Table 1-left. Results are presented in Table 2. It shows a clear improvement of the performance of semantic segmentation when using the proposed multi-task network. The best configuration is multi-task training using the raw depth map from SGBM. In Fig. 3, examples of semantic segmentations

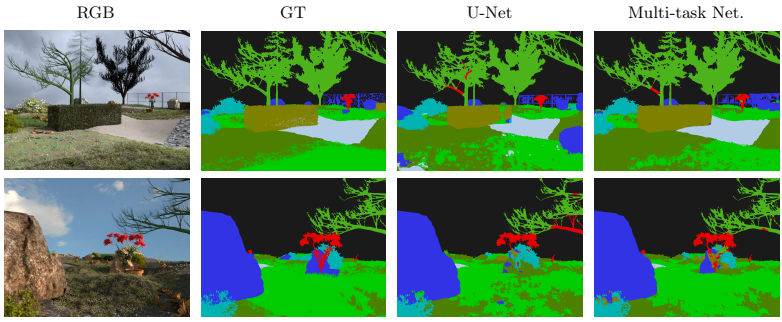


Fig. 3. Comparison of the semantic segmentation maps generated by U-Net trained on RGB images; and the proposed multi-task network architecture trained on RGB and raw depth images on input and extra refined depth on output.

of some 2D images from the dataset are displayed. It shows that co-learning enforces consistency with respect to the 3D structure. Indeed, neighbor pixels with the same depth (*i.e.* also close to each other in 3D) tend to get the same semantic label.

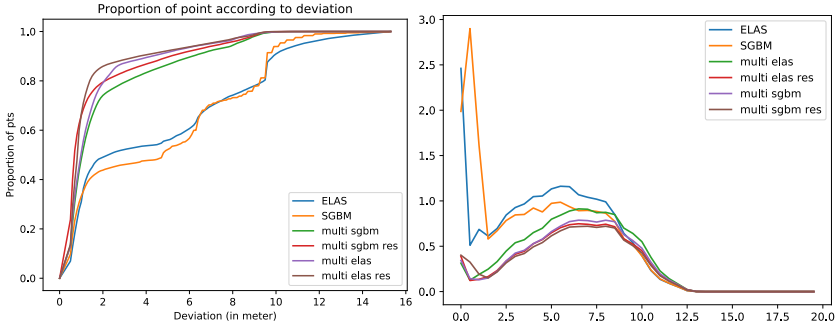
4.2 Depth estimation

The quality of 3D reconstruction highly depends on the estimation of an accurate depth map. In this paper, we propose to generate a precise depth map by refining a raw one obtained with a stereo pair. This process is one of the tasks of our multi-task network. In the following, we compare the performances of depth estimation using traditional stereo methods such as ELAS [31] and SGBM [32] with the performances of the refined depth estimates. We also evaluate the performance of a state-of-the-art single-image depth estimation approach, referred to as D3-Net [27].

The various depth map predictions are first compared using standard error measurements previously used for the same purpose [34,24] and defined in Table 1(right). We also provide the proportion of points with a deviation less than a given value in Fig. 4(a) and the RMS function with respect to the ground truth distance in Fig. 4(b). Several conclusions can be drawn from this. First, refinement of the geometric depth map using a multi-task neural network highly improves the depth estimation accuracy. Indeed for geometric approaches, only 40 % of the points have a deviation lower than 2m, while it reaches 80 % using the proposed multi-task approaches. One can note that improvement is specifically significant for small depth range, between 0 to 5m, which is crucial for safe autonomous navigation. Second, all multi-task learning approaches show similar results, with slightly better performances when the raw input comes from SGBM. Furthermore, our tests also show that using a state-of-the-art FCN for

Table 3. Comparisons of error metrics for depth estimation using geometric, D3-Net depth prediction and multi-task learning on the 3DRMS 2018 dataset.

	Methods		Error↓ **Test 0001**		Error↓ **Test 0224**	
	Input	Output	rel	rms	rel	rms
Geometric						
ELAS	RGBx2	D	0.526	2.140	0.444	1.993
SGBM	RGBx2	D	0.518	1.801	0.439	1.745
Mono image						
D3-Net Mono	RGB	D	0.145	0.755	0.110	0.477
Refinement						
FuseNet	RGB+D_{SGBM}	D	0.057	0.395	0.074	0.454
Multi-task Net.	RGB+D_{ELAS}	D+S	0.079	0.410	0.066	0.414
Multi-task Net.	RGB+D_{SGBM}	D+S	0.082	0.394	0.089	0.436



(a) (b)

Fig. 4. Comparison of performances of geometric methods (ELAS and SGBM) and the proposed approach, depth estimation through co-learning: (a) Proportion of 3D points with deviation less than a given value; (b) RMS of the depth estimates with respect to the ground truth distances.

single-image depth estimation outperforms the purely geometric approaches according to these standard, global metrics. As discussed in the following, this result can be explained by a better depth map segmentation obtained by deep learning approach.

Figure 5 shows examples of depth maps obtained with the various geometric or multi-task approaches. A geometric method such as SGBM results in accurate depth estimates but with a low quality segmentation of the depth map. On the contrary, a deep learning approach such as D3-Net shows an excellent depth segmentation, but produces biased depth values. Finally, the proposed approach which benefits from both geometrical and deep learning techniques shows the best results both in terms of accuracy and quality of depth segmentation.

4.3 Reconstruction

The reconstruction is performed with OpenChisel [33]. In this section, we present the final results of the reconstruction for both test sets 001 and 224 of the 3DRMS dataset. We evaluate the geometric quality of the reconstruction according to the

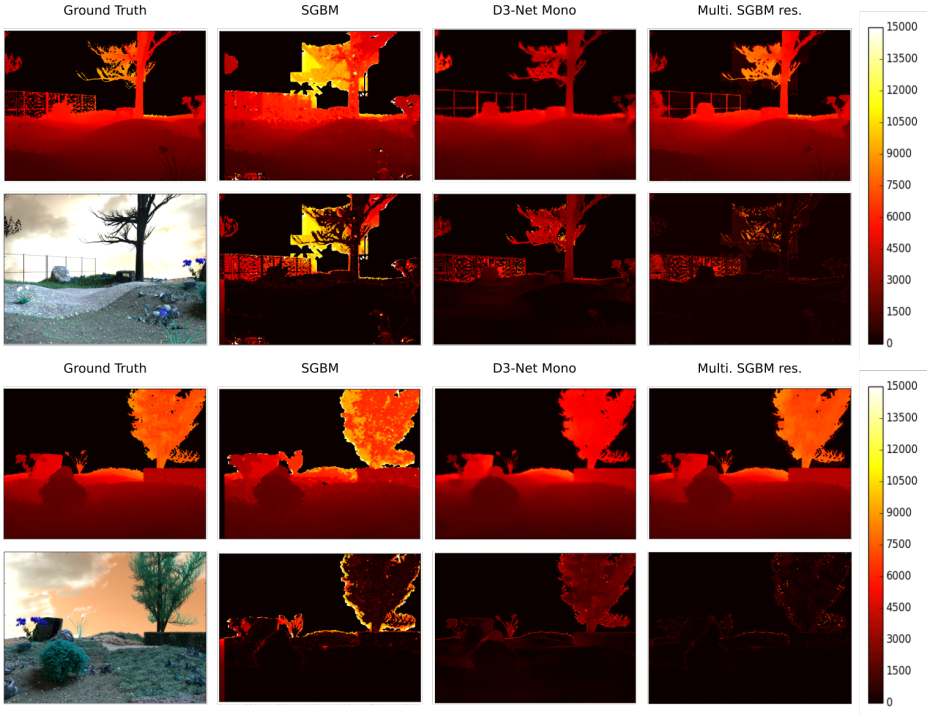


Fig. 5. Comparison of various depth estimation approaches: geometric approach (SGBM); depth prediction (D3-Net); and the proposed multi-task network (last column). *First row / Last row: depth maps / error maps in mm.*

depth map filtering strategy. We also provide quantitative and qualitative results on the semantics in 3D.

Geometric reconstruction As defined in [15], the quality of the reconstruction can be evaluated from two points of view. First, each point of the ground truth must be close to a point of the reconstructed scene, this is the completeness of the reconstruction, *i.e.*. it express how well the whole scene has been discovered and reconstructed. Second, each point of the reconstruction must be close to a point of the ground truth, this is accuracy. The accuracy aims at evaluating how well the reconstruction fits to the ground truth. In practice a good reconstruction is a compromise between completeness and accuracy; filling the space with points would improve the completeness while selecting only few points, well positioned, would improve the accuracy.
We use the following metrics for quantitative results (definition in table 4):

– *from ground truth to reconstruction*: the average distance of GT point to the mesh, and the completeness (the distance d such that 90% of the GT points are at distance less than d to the reconstruction).

Table 4. Error measurements used for geometric reconstruction quality estimation.

Metric	Definition
Average distance for A $\rightarrow$ B	$\frac{1}{\lvert A\rvert}\sum_{p\in A}\min_{q\in B} d(p,q)$
Accuracy % $<$ 5cm for A $\rightarrow$ B	$\frac{100}{\lvert A\rvert}\sum_{p\in A} 1_{\min_{q\in B} d(p,q)<0.05}$
Completeness dist 90% for A $\rightarrow$ B	$\min_d \frac{100}{\lvert A\rvert}\sum_{p\in A} 1_{\min_{q\in B} d(p,q)<d} = 90$

Note: 1 *is the indicator function.*

Table 5. Evaluation of the reconstruction on the 3DRMS dataset.

Filtering method	range	Full scene GT $\longrightarrow$ Recons. Av. dist.	Full scene GT $\longrightarrow$ Recons. Complet. dist. 90%	Full scene Recons. $\longrightarrow$ GT Av. dist.	Full scene Recons. $\longrightarrow$ GT Acc. % $<$ 5cm	Cropped scene z=1m GT $\longrightarrow$ Recons. Av. dist.	Cropped scene z=1m GT $\longrightarrow$ Recons. Complet. dist. 90%	Cropped scene z=1m Recons. $\longrightarrow$ GT Av. dist.	Cropped scene z=1m Recons. $\longrightarrow$ GT Acc. % $<$ 5cm
No filtering	5m	0.061	0.145	0.201	32.2%	-	-	-	-
	10m	0.061	0.164	0.311	20.9%	-	-	-	-
Gradient	5m	0.077	0.208	0.037	77.6%	0.042	0.102	0.027	86.3%
	10m	0.058	0.156	0.047	70.5%	0.043	0.109	0.031	83.9%
Gradient	5m	0.128	0.427	0.024	87.2%	0.052	0.134	0.020	90.7%
+ Erosion	10m	0.113	0.356	0.028	85.1%	0.052	0.134	0.022	89.8%

– *from reconstruction to GT*: the average distance of mesh vertices to GT, and the accuracy (percentage of vertices at distance less than 5cm to the GT)

We compute these metrics using CloudCompare [1]. For an easier readability, we restrict the numbers to the test set 001, results on the 224 scene being consistent the previous ones, the prediction method if the method using residual depth output with SGBM data as input.

We compare the effect of the filtering policies. We made reconstruction experiments with the two filtering methods and the baseline (no filtering) for OpenChisel clipping range r in $[5, 10]$ meters.

Table 5 presents these results for ranges 5m and 10m. The results are first computed with the full scene ground truth (including complete trees) and then with a cropped ground truth at 1m height (corresponding to the use case of autonomous lawnmower). As expected, giving all the points OpenChisel leads to better completeness but produce a lot of reconstruction artifacts, in particular at transitions between objects or sky. Filtering these points based on gradient produce much better results according to outlier production while ensuring a good completeness. On the opposite, the harder filter given by the gradient and an erosion improves even more the accuracy but drastically reduced the number of 3D points (*e.g.* top of the bushes and topiaries) which impacts the completeness. Better performances are achieved using a a cropped ground truth. This is mostly due to the small baseline of the stereo images and the ground view, leading to missing or uncertain tree reconstruction.

The compromise between completeness and accuracy is illustrated on figure 6. It presents, on the left, a graph showing the average distance for completeness

[1] CloudCompare: `https://www.danielgm.net/cc/`

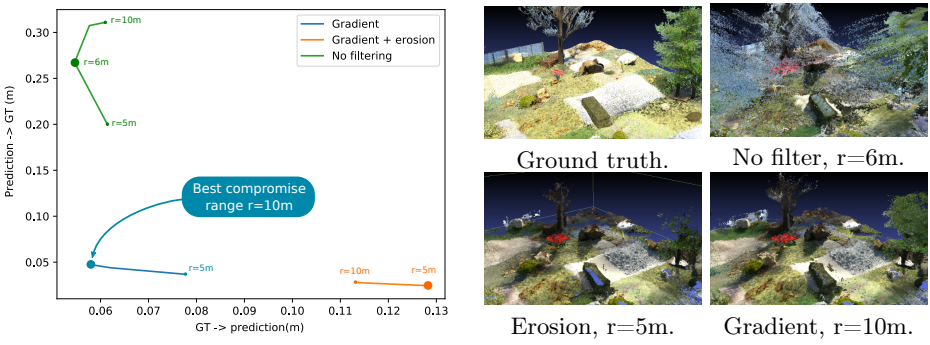


Fig. 6. Influence of the clipping range and the filtering method on 3D reconstruction.

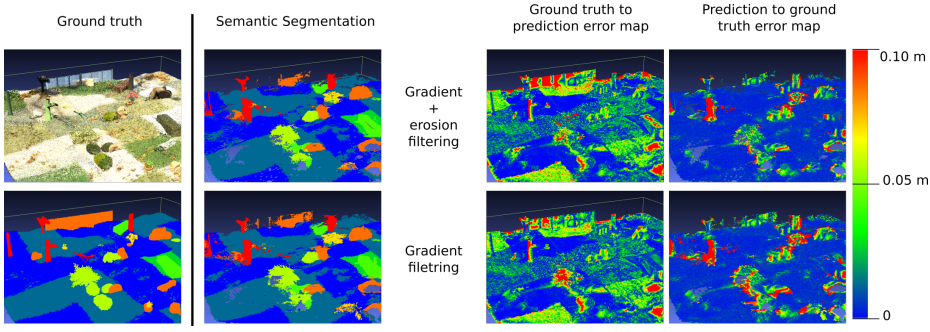


Fig. 7. Reconstructions for two filtering policies: Semantics mesh (left) and error heat maps(right).

function of the average distance for accuracy. The curves represent the evolution of theses distance with respect to the clipping range. The right part of the figure shows illustration for methods and ranges corresponding to the bigger dots in the previous graph.

Finally, error maps are presented on the left side of Fig. 7. For the $GT \rightarrow Predictions$ maps, the red points (error greater than 10cm) are the missing parts. For the $Predictions \rightarrow GT$ maps, red points correspond to hallucinated objects, particularly multiple tree trunks or flowers.

Semantics

Evaluation of 3D semantics is not straightforward: there is no direct correspondence between points of the ground truth and the reconstructed mesh. We use a strategy to create a geometric clone of the ground truth and assigning to each point the label of the nearest vertex of our reconstructed mesh. Thus, we have prediction/GT label pairs usable for metric computation. Table 6 presents the results for the *Multi-task Net.* with gradient filtering for overall accuracy (OA), average accuracy (Av Acc.) and average intersection over union (Av. IoU).

Table 6. Evaluation of the 3D semantics.

			Full scene			Cropped scene z=1m		
Method	range	Dataset	OA	Av. Acc.	Av. IoU	OA	Av. Acc.	Av. IoU
Gradient	10m	001	0.8950	0.8735	0.7285	0.8640	0.8705	0.7339

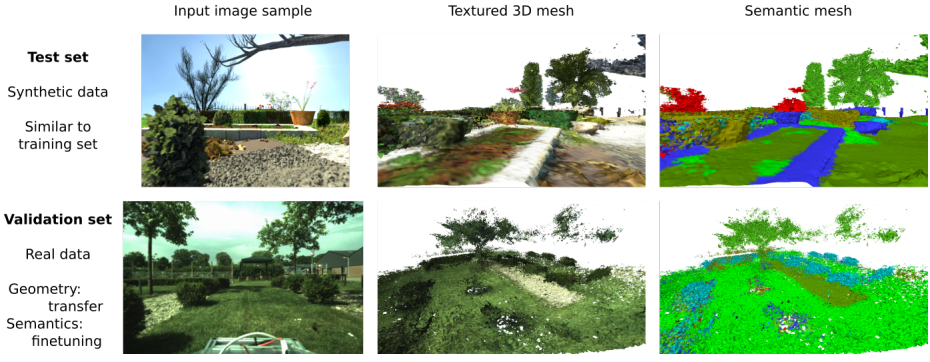


Fig. 8. Reconstructions on the tests sets of the 3DRMS datasets 2017 and 2018.

Left side of figure 7 shows snapshots of the surface with semantics labels. Most of the errors are located on the ground, mostly mixed grass and ground. The semantic prediction tend to produce smooth segmentations and fail to create very small connected component, such as pebbles in the grass.

Transfer to real data

The ultimate goal of the reconstruction pipeline is to be applicable to real data. To test the ability of our pipeline to generalize from the synthetic dataset to real outdoor data, we experiment the 3DRMS 2017 test dataset. The results are shown on Fig. 8. For visual comparison, we confront the reconstruction for the synthetic test (first row) and real data (second row), note the high difference between the two sample images. We tested first direct transfer of the neural network to the new dataset. While depth estimation was still efficient (middle image), semantic segmentation was deteriorated. To address this problem, we finetuned the segmentation decoder of the network on the train set of the 3DRMS 2017 dataset for ten epochs. Note that, in order to maintain the depth estimation quality, as the finetuning does not include depth ground truth, we froze the weights of the encoder and the depth decoder. Results are the in the right column: the semantics of the main objects and ground classes are well recovered.

4.4 Timings

We give some timing results for each separate block of our pipeline. The experiments were carried with an Intel Xeon CPU E3-1505M and Nvidia GTX1070

GPU.The stereo depth map estimation with SGBM takes 0.03s, the multi-task network depth and semantic inference takes 0.4s and the filtering step has a negligible computation time. For the 3D reconstruction, with a clipping range of 5m and with resolutions of 3cm, 5cm and 10cm, the related run times are respectively 0.4s, 0.15s and 0.1s per depth map. Higher clipping ranges increases the computation load as it significantly augment the number of points to use in the reconstruction. With a range of 10m and a resolution of 3cm, the integration of one depth map takes 2.2s. As our pipeline is designed to process incoming data online, one can expect each stereo pair to be processed in less than 0.85s for a high resolution map (3cm voxels - 5m range) and in less than 0.5s for lower resolution maps ($\geq$ 5cm voxels - 5m range), which are often sufficient for autonomous navigation.

5 Conclusion

In this paper, we have presented a 3D reconstruction approach from multiple stereo image pairs. The reconstruction pipeline mixes both the accuracy of geometric approaches and the complex, high-order modeling made possible by the deep neural networks. We show co-learning of depth estimation and pixelwise semantic labeling is possible in robotics scenarios and improves the framework at every stage. Indeed, the multi-task network, while being lighter than separate networks, is also more effective. The proposed approach is compatible with online mapping and does not require global optimization. The method has been evaluated on the 3DRMS 2018 simulated dataset.

A close look at the reconstructed surfaces shows that most of the geometric errors come from the duplication of some objects. Moreover, the main part of semantic errors are due to mis-detected pixels which deteriorate the global score while most of the othe objects have been correctly recognized. To improve these aspects of the method, future works will include performing object detection and tracking during the sequence. First, the object identification between images would reduce the number of instances in the final product and second, labels would be regularized at object level. Even though it would make our method less suitable for online use, we also examine the possibility of a posteriori spatial regularization on the reconstructed surface such as conditional random fields.

References

1. Tylecek, R., Sattler, T., Le, H.A., Brox, T., Pollefeys, M., Fisher, R.B., Gevers, T.: The second workshop on 3d reconstruction meets semantics: Challenge results discussion. In Leal-Taixé, L., Roth, S., eds.: ECCV 2018 Workshops, Cham, Springer International Publishing (2019) 631–644
2. Cole, D.M., Newman, P.M.: Using laser range data for 3D SLAM in outdoor environments. In: Proc. of IEEE International Conference on Robotics and Automation (ICRA). (May 2006) 1556–1563
3. Whelan, T., Salas-Moreno, R.F., Glocker, B., Davison, A.J., Leutenegger, S.: Elasticfusion: Real-time dense slam and light source estimation. The International Journal of Robotics Research
4. Mur-Artal, R., Montiel, J.M.M., Tardós, J.D.: ORB-SLAM: A Versatile and Accurate Monocular SLAM System. IEEE Transactions on Robotics (T-RO) (2015)
5. Forster, C., Zhang, Z., Gassner, M., Werlberger, M., Scaramuzza, D.: SVO: Semidirect Visual Odometry for Monocular and Multicamera Systems. IEEE Transactions on Robotics (T-RO) (2017)
6. Lynen, S., Sattler, T., Bosse, M., Hesch, J.A., Pollefeys, M., Siegwart, R.: Get out of my lab: Large-scale, real-time visual-inertial localization. In: Robotics: Science and Systems. (2015)
7. Schneider, T., Dymczyk, M.T., Fehr, M., Egger, K., Lynen, S., Gilitschenski, I., Siegwart, R.: maplab: An open framework for research in visual-inertial mapping and localization. IEEE Robotics and Automation Letters (2018)
8. Schönberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: CVPR. (2016)
9. Moulon, P., Monasse, P., Marlet, R., Others: Openmvg. an open multiple view geometry library. `https://github.com/openMVG/openMVG`
10. Civera, J., Gálvez-López, D., Riazuelo, L., Tardós, J.D., Montiel, J.M.M.: Towards semantic slam using a monocular camera. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). (Sept 2011) 1277–1284
11. McCormac, J., Handa, A., Davison, A., Leutenegger, S.: Semanticfusion: Dense 3d semantic mapping with convolutional neural networks. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). (2017)
12. Schönberger, J.L., Geiger, A., Pollefeys, M., Sattler, T.: Semantic visual localization. In: CVPR. (2018)
13. Hermans, A., Floros, G., Leibe, B.: Dense 3d semantic mapping of indoor scenes from rgb-d images. In: Proc. of Int. Conf. on Robotics and Automation. (2014)
14. Ma, L., Stueckler, J., Kerl, C., Cremers, D.: Multi-view deep learning for consistent semantic mapping with rgb-d cameras. In: IROS, Vancouver, Canada (2017)
15. Strisciuglio, N., Tylecek, R., Petkov, N., Bieber, P., Hemming, J., van Henten, E., Sattler, T., Pollefeys, M., Gevers, T., Brox, T., Fisher, R.: Trimbot2020: an outdoor robot for automatic gardening. In: Proc. of International Symposium on Robotics. (2018)
16. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence (2017)
17. Guerry, J., Boulch, A., Le Saux, B., Plyer, A., Moras, J., Filliat, D.: SnapNet-R: Consistent 3d multi-view semantic labeling for robotics. In: Proc. of Int. Conf. of Comp. Vis. Workshop on 3D Reconstruction meets Semantics (ICCVW), Venice, Italy (2017)

18. Taguchi, Y., Feng, C.: Semantic 3D reconstruction using depth and label fusion. In: Proc. of Int. Conf. of Comp. Vis. Workshop on 3D Reconstruction meets Semantics (ICCVW), Venice, Italy (2017)
19. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2015) 3431–3440
20. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. MICCAI (2015)
21. Hazirbas, C., Ma, L., Domokos, C., Cremers, D.: Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. In: Proc. of Asian Conference on Computer Vision (ACCV), Springer (2016) 213–228
22. Audebert, N., Le Saux, B., Lefèvre, S.: Semantic Segmentation of Earth Observation Data Using Multimodal and Multi-scale Deep Networks. In: Asian Conference on Computer Vision (ACCV16), Taipei, Taiwan (November 2016)
23. Qi, X., Liao, R., Jia, J., Fidler, S., Urtasun, R.: 3d graph neural networks for rgbd semantic segmentation. In: Proc. of Int. Conf. on Computer Vision (ICCV). (2017)
24. Eigen, D., Fergus, R.: Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-Scale Convolutional Architecture. ICCV (2015)
25. Xu, D., Ricci, E., Ouyang, W., Wang, X., Sebe, N.: Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. arXiv preprint arXiv:1704.02157 (2017)
26. Kendall, A., Gal, Y.: What uncertainties do we need in bayesian deep learning for computer vision? arXiv preprint arXiv:1703.04977 (2017)
27. Carvalho, M., Saux, B.L., Trouvé-Peloux, P., Almansa, A., Champagnat, F.: On regression losses for deep depth estimation. In: ICIP. (2018)
28. Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., Navab, N.: Deeper depth prediction with fully convolutional residual networks. In: 3D Vision (3DV), 2016 Fourth International Conference on, IEEE (2016) 239–248
29. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. (2016) 770–778
30. Huang, G., Liu, Z., Weinberger, K., van der Maaten, Laurens: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017)
31. Geiger, A., Roser, M., Urtasun, R.: Efficient large-scale stereo matching. In: ACCV. (2010)
32. Hirschmueller, H.: Stereo processing by semiglobal matching and mutual information. PAMI (2008)
33. Klingensmith, M., Dryanovski, I., Srinivasa, S., Xiao, J.: Chisel: Real time large scale 3d reconstruction onboard a mobile device. In: Robotics Science and Systems 2015. (July 2015)
34. Li, B., Shen, C., Dai, Y., van den Hengel, A., He, M.: Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs. In: CVPR. (2015) 1119–1127