SEMANTIC SEGMENTATION OF TERRESTRIAL LIDAR DATA USING CO-REGISTERED RGB DATA

E. Sanchez Castillo[1]*, D. Griffiths[1], J. Boehm[1]

[1]Dept. of Civil, Environmental & Geomatic Engineering, University College London,
Gower Street, London, WC1E 6BT UK - (erick.castillo.19, david.griffiths.16, j.boehm)@ucl.ac.uk

Commission II, WG II/3

KEY WORDS: terrestrial laser scanning, point cloud, panoramic image, semantic segmentation, convolutional neural network

ABSTRACT:

This paper proposes a semantic segmentation pipeline for terrestrial laser scanning data. We achieve this by combining co-registered RGB and 3D point cloud information. Semantic segmentation is performed by applying a pre-trained off-the-shelf 2D convolutional neural network over a set of projected images extracted from a panoramic photograph. This allows the network to exploit the visual image features that are learnt in a state-of-the-art segmentation models trained on very large datasets. The study focuses on the adoption of the spherical information from the laser capture and assessing the results using image classification metrics. The obtained results demonstrate that the approach is a promising alternative for asset identification in laser scanning data. We demonstrate comparable performance with spherical machine learning frameworks, however, avoid both the labelling and training efforts required with such approaches.

1. INTRODUCTION

Over the past decade, the construction and real estate sectors have increasingly used Terrestrial Laser Scanners (TLS) to capture and document building interiors. This process usually delivers a dense, high-quality point cloud, which can serve as the basis for remodelling and asset management. Furthermore, modern instruments not only capture the 3D positions of interior surfaces, but also colour information from panoramic photographs, making it possible for a point cloud to be reasoned from both its spatial and photometric qualities. A key task in point cloud scene understanding is assigning an object label for every point, often referred as either *per-point classification* or *semantic segmentation*. In this work we adopt the latter.

In recent years, a surge of deep learning approaches for point cloud semantic segmentation have been proposed. Nevertheless, the problem is still considered hard. This can be accredited to a number of reasons. Firstly, point clouds are typically unordered, and sparse data types. This prevents normal convolution kernels, which assume discrete structured data, from being effective. As a result, deep learning based 2D approaches typically remain more mature. Despite great progress in addressing this problem (Qi et al., 2017b; Hermosilla et al., 2018; Thomas et al., 2019), another issue looms. Modern deep learning based methods require very large labelled datasets, however, such datasets for 3D data are typically not available at the same scale as that for their 2D counterparts.

In light of such limitations, we instead ask the question, can 3D point cloud semantic segmentation be achieved using only 2D models? Ultimately allowing us to exploit existing 2D CNN architectures and massive manually labelled 2D datasets.

In answering this, we propose a methodology which projects 3D data with co-registered RGB data into 2D images which can be consumed by standard 2D Convolutional Neural Networks (CNNs). Our multi-stage pipeline first starts with the extraction of a panoramic image from a TLS acquired point cloud. Next, we compute tangential images in a perspective projection which can be fed into a CNN to map RGB values to per-pixel labels. Finally, we project the label map back to the point cloud to obtain per-point labels. Through a hyperparameter grid search we find that our method can be used to obtain a competitive semantic segmentation of point clouds leveraging only a pre-trained off-the-shelf 2D CNN without any additional labelling or domain adaptation.

Empirically we show that despite the raw image data being in an equirectangular projection, CNNs trained using the more common rectilinear projection produce respectable labels using our approach. Our pipeline therefore makes data captured by polar devices, such as a TLS, compatible with any standard CNN-based image segmentation architecture.

2. RELATED WORKS

The process of assigning per-point classification labels to point clouds has a rich history. Traditionally, success has been owed to supervised machine learning based techniques. As a single point does not contain enough information to determine its label, researchers explore methods to encompass local neighbourhood context. Demantké et al. (2011), Weinmann et al. (2015) and others demonstrated the effectiveness of explicitly encoding features computed from a points local neighbourhood. Features such as linearity, planarity and Eigenentropy are calculated for each point and passed into a Random Forest classifier. This can be performed at scale (Liu and Boehm, 2015). Other feature sets such as Fast Point Feature Histograms (FPFH) (Rusu et al., 2009) and Color Signature of Histogram of Orientations (SHOT) (Salti et al., 2014) have also shown promising results.

More recently, there has been a surge of deep learning based approaches (Griffiths and Boehm, 2019). The seminal work of

* Corresponding author

PointNet (Qi et al., 2017a) demonstrated the compatibility of deep learning with such problems. However, PointNet did not exploit local neighbourhood features like those explicitly encoded in early works. PointNet++ (Qi et al., 2017b) showed that by combining a PointNet with local neighbourhood grouping and sampling module, results could be significantly improved. More recent research looks at developing convolution kernels (which experienced unprecedented success in the 2D domain) that are capable of working in the unordered, sparse and continuous domain where the point cloud exists. Examples such as Monte Carlo Convolutions (Hermosilla et al., 2018), Kernel Point Convolutions (Thomas et al., 2019) and PointConv (Wu et al., 2019) address this.

In the 2D domain researchers have developed methods for processing spherical images. For example, the spherical cross-correlation and generalised Fourier transform algorithms in Cohen et al. (2018), the adaptation of different convolution layers in Yu and Ji (2019), or transforming encoders and decoders for understanding the geometry variance derived from the input equirectangular panoramic image in Zhao et al. (2018b). Zhao et al. (2018a) improved spherical analysis for equirectangular images by creating networks that can iterate between image sectors and classify panoramas with significant performance and speed, which is comparable to classic two-dimensional networks.

As it is possible for 3D point clouds to be projected into a 2D spherical domain, naturally, approaches have been proposed to exploit the spherical 2D CNNs for 3D semantic segmentation. Jiang et al. (2019), parse spherical grids approximated to a given underlying polyhedral mesh, using what the author calls "Parameterised Differential Operators", which are linear combinations of differential operators that avoid geodetic computations and interpolations over the spherical projection. Similarly, Zhang et al. (2019) propose an orientation-aware semantic segmentation on icosahedral spheres.

Concurrent research has also been present in the autonomous driving domain. Wu et al. (2018); Wang et al. (2018) transform 3D scanner data into 2D spherical image which is fed into a 2D CNN, before unprojecting labels back to the original point cloud. These methods are typically a lot faster than purely 3D approaches as projection and 2D convolutions are much faster than 3D neighbourhood searches required by geometric-based approaches. Similar to our work, Tabkha et al. (2019) perform semantic segmentation using a Convolutional Neural Network (CNN) on RGB images derived by projecting coloured 3D point clouds. However, our work differs from these approaches as we do not use an unordered point cloud as the representation for the LiDAR data. Instead, we use the ordered panoramic representation that is generated by polar measuring devices such as TLS. On the downside this restricts our approach to single scans captured with static TLS and excludes e.g., mobile scanners.

Also similar to our work, Eder et al. (2020) divide a spherical panoramic image into tangential icosahedral planes and the project individual perspective images. This allows each image to be fed into a pre-trained 2D semantic segmentation CNN. Furthermore, Eder et al. (2020) obtained comparable results using standard CNNs to more specialised spherical CNNs.

3. METHODOLOGY

Given a point cloud $\mathcal{P} \in \mathbb{R}^{n \times k}$ captured using a polar-based TLS scanner, we aim to assign a per-point object class label i.e. $\mathbb{R}^{n \times k} \rightarrow \mathbb{R}^{n \times 1}$ where n is the number of points in $\mathcal{P}$ and $k \in \mathbb{R}^{x,y,z,r,g,b}$ (although k can include other sensor features such as intensity). Whilst in remote sensing and photogrammetry this problem is typically referred to as (per-point) *classification*, we use the term *semantic segmentation* common in image processing as these are the networks we use for creating the label mapping function $f : \mathbb{R}^{n \times k} \rightarrow \mathbb{R}^{n \times 1}$.

Our methodology can be split into the following primary processes. First, a point cloud $\mathcal{P}$ with corresponding RGB image data $\mathcal{I} \in \mathbb{R}^{h \times w \times 3}$ is captured using a survey-grade TLS. Such scanners are two-axis polar measurement instruments and acquire quasi regular samples on the two axes, effectively creating a regular grid in the polar space. This representation is also commonly used in panoramic imaging and is referred to as equirectangular. The scanner hardware or associated software warps the image data captured alongside the point cloud into this projection. The resulting panoramic colour images can be extracted using open standard file formats.

Next, we convert the information of the panoramic image $\mathcal{I}$ to tangential images $\mathcal{I}^T$ to simulate a rectilinear lens. This projection is not a valid transformation for the complete panoramic image, and therefore we create a sequence of overlapping partial images. The position of the tangential images is determined using spherical grid sequence intervals, creating an almost equal distribution over the spherical space such that $\mathcal{I}^T = \{\mathcal{I}_1^t \dots \mathcal{I}_n^t\}$. We then obtain per-pixel labels $\mathcal{I}^s \in \mathbb{R}^{h \times w \times 1}$ by utilising a semantic segmentation CNN S such that $\mathcal{I}_i^s = S(\mathcal{I}_i^t)$. All partial rectilinear label images $\mathcal{I}_i^s$ are then projected back to the original panoramic projection, allowing the final label map $\mathcal{I}^C$ to be created using the confidence scores obtained by the semantic segmentation process. In our experiments S is a pre-trained UperNet model (Xiao et al., 2018) which was trained on the rectilinear based ADE20K dataset (Zhou et al., 2017). Finally, we map the class labels $\mathcal{I}^C \rightarrow \mathcal{P}$ using the co-registration matrix, assigning per-point labels. Figure 1 gives a graphical overview of the process. In the following sections we will discuss each stage in detail.

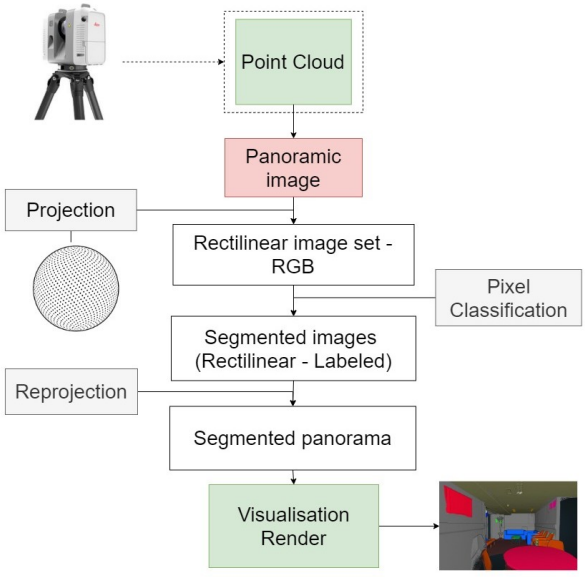

Figure 1. This diagram shows the proposed semantic segmentation process using panoramic images from TLS data.

3.1 Data acquisition

Our scanner data ($\mathcal{P}$ and $\mathcal{I}$) in this project was collected using a Leica RTC360 TLS. This system (along with many other commercially available systems) captures 3D measurements in a structured sequence. As mentioned above it acquires the points over a quasi-regular grid in the polar space. This polar grid is directly represented as a two-dimensional matrix. This enables the projection of 3D point cloud data from a polar to an equirectangular projection. Effectively transforming the captured data into a panoramic image (Figure 2 Row 1). This representation of TLS data is long established for image processing and object extraction (Boehm and Becker, 2007; Eysn et al., 2013).

We processed all data with the manufacturer software, exporting the point cloud to a grid-type separator file format that preserves the orientation header of the scan position and each corresponding scanned point on the ordered grid. We utilise this raster grid to extract the panoramic image directly. The final resolution of our panoramic image is $(20,334 \times 8,333)$. This is generated from a maximum of $169,443,222$ points (as limited by the TLS), however, in practice much fewer points are actually captured due to lack of returns from angular surfaces, windows etc.

3.2 Rectilinear projection

With the TLS capture described above having a spherical equidistant subdivision, the creation of an equirectangular projection is trivial, interpreting the data as a raster. As this projection is neither equal-area nor conformal, there are distortions in the resulting panoramic image. To address the spherical distortion, we need to define a rectilinear projection for tangential images and a subdivision method from where the tangential points will be defined for each individual projection.

The mathematical foundations used in this reprojection process are detailed as follows, extracted from Weisstein (2018). Given a point $p_i \in \mathcal{P}$ with a latitude and longitude (λ, ϕ), the transformation equations for the creation of a tangent plane at that point, with a projection with central longitude λ_0 and central latitude ϕ_1 are given by:

$$x = \frac{\cos\phi \sin(\lambda - \lambda_0)}{\cos c} \tag{1}$$

$$y = \frac{\cos\phi_1 \sin\phi - \sin\phi_1 \cos\phi \cos(\lambda - \lambda_0)}{\cos c} \tag{2}$$

Where c is the angular distance of the point (x, y) from the projection centre, given by:

$$\cos c = \sin\phi_1 \sin\phi + \cos\phi_1 \cos\phi \cos(\lambda - \lambda_0) \tag{3}$$

Knowing the image size and the corresponding field of view (FOV) angle for the respective c, we can generate individual images $\mathcal{I}^T$ from the full-dome panorama $\mathcal{I}$. The latitude and longitude (λ, ϕ) positions of the spherical intervals are defined by the golden ratio angle separation, where the generative spirals of a Fibonacci lattice turn between consecutive points along a symmetrical spiral sphere (Gonzalez, 2009).

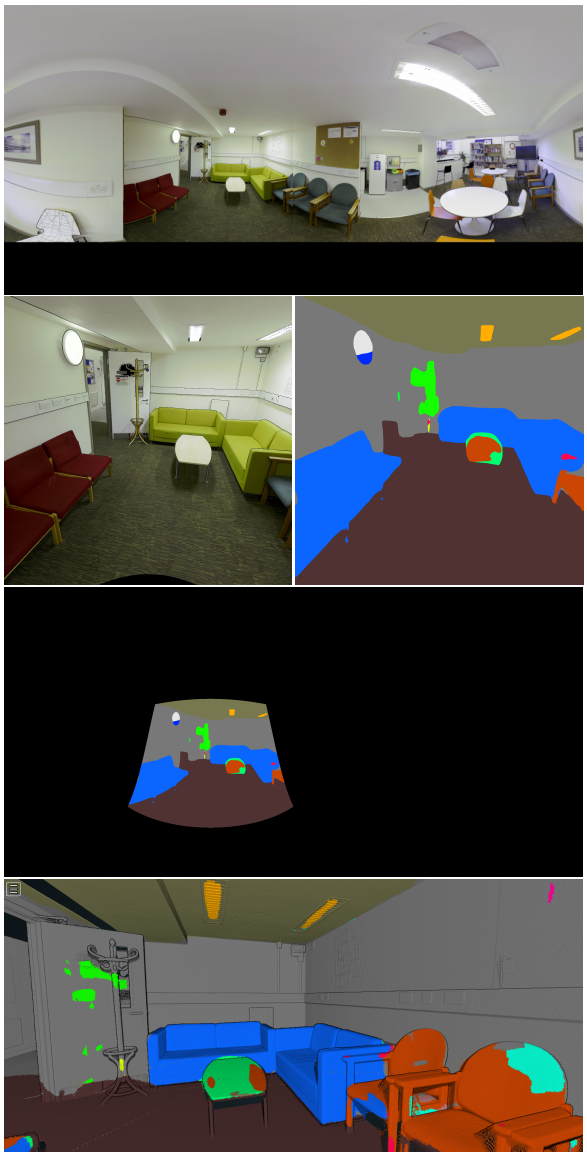

Figure 2. Data at different stages of the pipeline. **Row 1** Panorama image created from TLS point cloud with the co-registered RGB information. **Row 2** Example of tangential image in rectilinear projection and segmentation result. **Row 3** The semantic segmentation output re-projected from tangential to equirectangular. The full map is given in Figure 6. **Row 4** Point cloud rendering with labels from merged equirectangular segmentation map.

To create the lattice, the function of this sequence for the symmetrical points n is described as $n = 2N + 1$ where N is any natural number defining the desired interval subdivision and the integer i range from $-N$ to $+N$. The spherical coordinates of ith point are:

$$\mathrm{lat}_i = \arcsin\left(\frac{2i}{2N+1}\right) \tag{4}$$

$$\mathrm{lon}_i = 2\pi i \Phi^{-1} \tag{5}$$

where:

$$\Phi = 1 + \Phi^{-1} = (1 + \sqrt{5})/2 \simeq 1.618 \tag{6}$$

and Φ is the golden ratio.

The result of this projection, which is also referred to as gnomonic projection, is a quasi perspective image $\mathcal{I}_i^t$ (Figure 2 Row 2) which is equivalent to an image captured by a camera with a rectilinear lens. Typical cameras available today try to achieve such a projection. As a result the projected images are of the same projection as those of most large scale benchmark datasets used to train 2D ML models.

3.3 Semantic segmentation

At the centre of our pipeline is a deep learning based semantic segmentation network S which maps a single tangential image to a probability class map $S : \mathcal{I}_i^T \rightarrow \mathcal{I}_i^s$ (Figure 3). A key benefit of our pipeline is that it is compatible with any choice of S. In such a fast moving field this allows the user to drop in the current best performing network implementation. In this work we opt for the widely used UPerNet network (Xiao et al., 2018) as an example.

We choose this network for several reasons. Firstly, the network performs competitively on computer vision benchmarks. Next, the authors offer an easy-to-use publicly available implementation. Lastly, the authors release pre-trained weights on the ADE20K indoor scene parsing dataset, which contains all of the objects present in our datasets.

We note, that whilst any 2D CNN semantic segmentation network can be used in our pipeline, it is important that the user also has access to the prediction confidence scores $\mathcal{C}_i^s \in \mathcal{C}^C$ (as output from the final prediction probability distribution). These values are used to handle redundant label when recomposing $\mathcal{I}_i^s \in \mathcal{I}^C \rightarrow \mathcal{I}^C$. This is discussed in detail in Section 3.4.

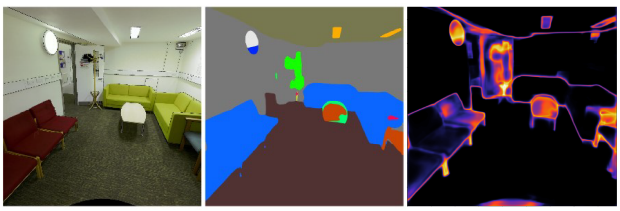

Figure 3. Projected tangential image (**left**), visualisation of UPerNet semantic segmentation classes (**centre**) and corresponding confidence map where darker is more confident (**right**).

3.4 Reprojection

After obtaining the semantically segmented images $\mathcal{I}_i^s \in \mathcal{I}^C$ and the confidence matrix associated to each tangential position (λ, ϕ), it is necessary to warp back the images to the equirectangular projection, in order to obtain a new set of panoramic images for the posterior unification process. The inverse transformation equations, having a pixel coordinate (x, y), are given by:

$$\phi = \sin^{-1}\left(\cos c \sin\phi_1 + \frac{y \sin c \cos\phi_1}{\rho}\right) \tag{7}$$

$$\lambda = \lambda_0 + \tan^{-1}\left(\frac{x \sin c}{\rho \cos\phi_1 \cos c - y \sin\phi_1 \sin c}\right) \tag{8}$$

With the central longitude λ_0, central latitude ϕ_1, ϕ and λ being the resulting latitude and longitude for each reprojected pixel (x, y), respectively. ρ and c are defined as:

$$\rho = \sqrt{x^2 + y^2} \tag{9}$$

$$c = \tan^{-1}\rho \tag{10}$$

The resulting image has the corresponding order of latitude and longitude of the spherical subdivision (Figure 2 Row 3).

3.5 Panoramic Label Map

Following the processing methodology, it is necessary to recreate a full resolution panoramic label image $\mathcal{I}^C$ from the overlapping tangential semantic segmentation maps (i.e. $\mathcal{I}_i^t \in \mathcal{I}^C \rightarrow \mathcal{I}^C$). To achieve this, we adopt a winner-take-all approach from the corresponding pixel confidence scores $\mathcal{C}^s \in \mathcal{C}^C$. The final output map for any redundant pixels is therefore:

$$\mathcal{I}^C(\lambda, \phi) = \max_{i,j}[\mathcal{C}_i^s(\lambda, \phi), \mathcal{C}_j^s(\lambda, \phi)] \tag{11}$$

3.6 Point cloud semantic segmentation

As a final step we map the equirectangular label map onto the original point cloud (i.e. $\mathcal{I}^C \rightarrow \mathcal{P}$). This is easily achieved by storing the original mapping $\mathcal{P} \rightarrow \mathcal{I}$ (Section 3.1). Using the reverse of this mapping we simply assign each point $p_i \in \mathcal{P}$ its corresponding value from $\mathcal{I}^C$. A rendering of the point cloud with label colours is shown in Figure 2 Row 4.

4. RESULTS

We test our methodology outlined in Section 3 for a range of configurations. Furthermore, we evaluate our approach on both an internal dataset and a sample from the common 2D3DS benchmark dataset (Armeni et al., 2017).

4.1 Performance metrics

It is important to define the metrics used to evaluate our proposed pipelines performance. Whilst the 2D3DS dataset contains labels, our internal dataset did not. It is therefore necessary to label the ground truth data. As we are not using the dataset for training a CNN, all data is test data, and as such, we do not require a large dataset. All data was therefore manually annotated using standard image processing software with a graphical user interface.

To evaluate each scenario's performance, we opt for the widely used Intersection over Union metric (Everingham et al., 2010), averaged over all classes (mIoU). In practise we compute the IoU over the $N \times N$ confusion matrix C, where N is number of classes (21 in our case). Let c_{ij} be a single entry in C, where c_{ij} is a number of sampled from the ground truth class i predicted as class j, then the per-class IoU can be computed as:

$$IoU_i = \frac{c_{ii}}{c_{ii} + \sum_{j \neq i} c_{ij} + \sum_{k \neq i} c_{ki}} \tag{12}$$

$mIoU$ is then:

$$mIoU = \frac{1}{N}\sum_{i=1}^{N} IoU_i \tag{13}$$

In addition to $mIoU$ we also compute an average of the overall accuracy, however, as accuracy can be non-robust when strong class imbalance is present, we treat $mIoU$ as our primary metric. Nevertheless, we compute $mAcc$ as:

$$mAcc = \frac{\sum_{i=1}^{N} c_{ii}}{\sum_{j=1}^{N}\sum_{k=1}^{N} c_{jk}} \tag{14}$$

4.2 Hyperparameter search

It is evident that the configuration used to perform $\mathcal{I} \rightarrow \mathcal{I}^T$ can affect model performance. We therefore perform a hyperparameter grid search to find the optimum configurations for generating the tangential images $\mathcal{I}^T$ with respect to our performance metrics. We select the following hyperparameters for optimisation; `spherical tangent points location`, `fov`, `image size` and `image ratio`. Results of the search are visualised in Figure 4.

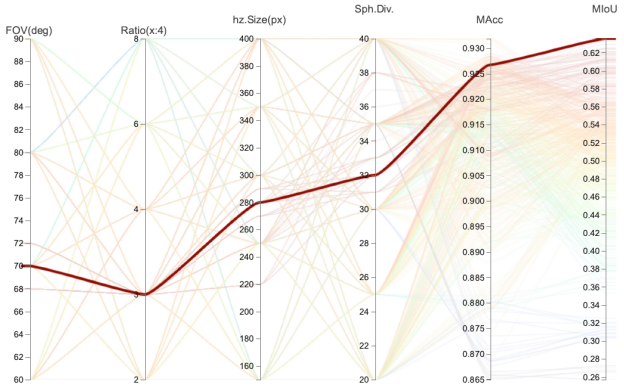

Figure 4. Results of the hyperparameter's optimisation process (Section 4.2).

4.3 Internal dataset

The result of the $mIoU$ evaluation shows that the 70-degree field of view, a 3:4 aspect ratio, an image size of 840×1120 and a spherical subdivision with 32 tangential points is the optimal pipeline configuration for this dataset, as shown in Figure 4. It is also remarkable that an increase in the resolution of the tangential images does not improve the final performance. Additionally, greater redundancy in the spherical positions also results in a decreased performance.

The final semantic segmentation image $\mathcal{I}^C$ with the optimum hyperparameters is shown in Figure 6 (top). Analysing the areas captured in the original panorama from the TLS visible in Figure 5 (top), versus the final segmented image, we note high precision is achieved at the object boundaries, especially on the furniture and walls. In addition, the $mIoU$ performance achieved is superior to the analysis of the raw panoramic image

Figure 5. Original panoramic images: Internal TLS capture **(top)** and 2D3DS panoramic RGB sample **(bottom)**

Table 1. Comparison of our proposed method using partial projections and applying the same CNN directly to the raw panorama with no projections.

Method	MAcc	MIoU
Raw panorama	0.912%	0.371%
OURS (TLS Dataset)	**0.927%**	**0.636%**
OURS (2D3DS Dataset)	**0.896%**	**0.472%**

$\mathcal{I}$, extracted directly from the point cloud $\mathcal{P}$. The final results with the optimum optimisation are shown in Table 1.

The normalised confusion matrix (Figure 7) demonstrates that our pipeline is able to identify the majority of the required classes presented in the panoramic scene. However, we note classes H (door) and I (desk) are poorly detected.

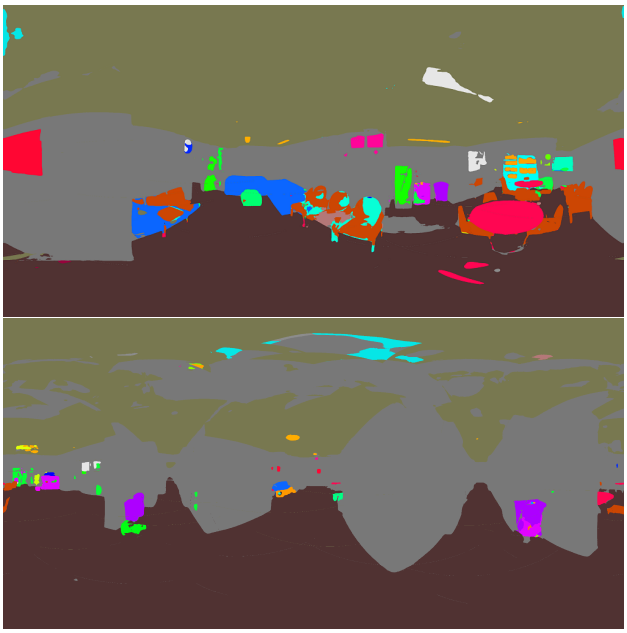

Figure 6. Final semantic segmentation results for our internal dataset (**top**) and 2D3DS dataset (**bottom**).

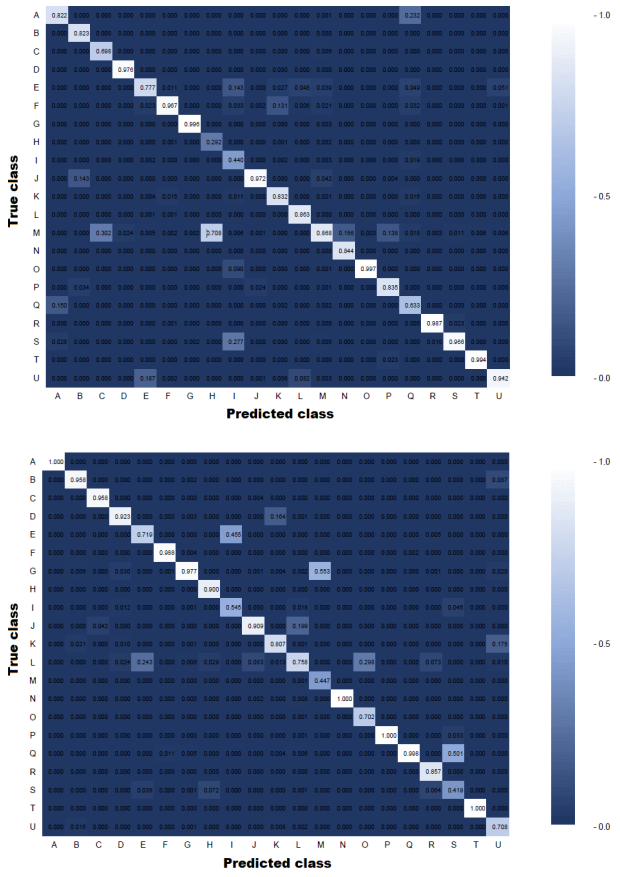

Figure 7. Normalised confusion matrix for our internal dataset (**top**) and 2D3DS dataset (**bottom**).

4.4 2D3DS dataset

We processed the selected 2D3DS panorama shown in Figure 5 (bottom), using the same methodology. We obtain the resulting shown in Figure 6 (bottom). The output segmentation map $\mathcal{I}^C$ is compared to the provided ground truth data. The selected image has a resolution of 4096×2048. The resulting image is generated by considering the best value obtained in the grid search presented before, but adjusting the image size and FOV resolution, with 80-degrees FOV, an aspect ratio of 3:4, an image resolution of 600×1200 and the spherical interval division as 32 tangential points.

Qualitatively analysing Figure 6 (bottom vs. top row), it is evident that the proposed method does not achieve similar performance in the lower resolution 2D3DS dataset, in comparison with the internal high-resolution TLS dataset. This is particularly evident for the ceiling. However quantitatively, it is clear from the confusion matrix (Figure 7 bottom) that nevertheless most areas of the dataset were correctly classified.

5. CONCLUSION

We presented a pipeline for semantic segmentation of TLS point clouds for indoor scenes. We show that by exploiting co-registered RGB image data, we can perform semantic segmentation using standard 2D CNNs. These labels can then be mapped back onto the original 3D point cloud data. We demonstrate satisfactory results using a pre-trained off-the-shelf 2D CNN, eliminating the need for manually labelled training data or specialised 3D point cloud networks. This allows us to exploit large 2D labelled datasets for 3D point cloud semantic segmentation. Furthermore, our results show that despite our original data being in an equirectangular projection, we still achieve reasonable class labels from a network trained on more commonly available rectilinear images. Whilst we expect results to improve if a network is trained directly on equirectangular images, we show that this is not strictly necessary. This significantly reduces workload and accelerates the adoption of new DL frameworks for TLS data.

ACKNOWLEDGEMENTS

This research was partially funded by the National Agency for Research and Development (ANID) of the Government of Chile, through the program ”Magister en el extranjero 2018” - 73190381.

References

Armeni, I., Sax, S., Zamir, A. R., Savarese, S., 2017. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*.

Boehm, J., Becker, S., 2007. Automatic marker-free registration of terrestrial laser scans using reflectance. *Proceedings of the 8th conference on optical 3D measurement techniques, Zurich, Switzerland*, 9–12.

Cohen, T. S., Geiger, M., Köhler, J., Welling, M., 2018. Spherical CNNs. *International Conference on Learning Representations*.

Demantké, J., Clément Mallet, N. D., Vallet, B., 2011. DIMENSIONALITY BASED SCALE SELECTION IN 3D LIDAR POINT CLOUDS. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38(5/W12).

Eder, M., Shvets, M., Lim, J., Frahm, J. M., 2020. Tangent images for mitigating spherical distortion. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12423–12431.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., Zisserman, A., 2010. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2), 303–338.

Eysn, L., Pfeifer, N., Ressl, C., Hollaus, M., Grafl, A., Morsdorf, F., 2013. A practical approach for extracting tree models in forest environments based on equirectangular projections of terrestrial laser scans. *Remote Sensing*, 5(11), 5424–5448.

Gonzalez, A., 2009. Measurement of Areas on a Sphere Using Fibonacci and Latitude–Longitude Lattices. *Mathematical Geosciences*, 42(1), 49. https://doi.org/10.1007/s11004-009-9257-x.

Griffiths, D., Boehm, J., 2019. A review on deep learning techniques for 3D sensed data classification. *Remote Sensing*, 11(12), 1499.

Hermosilla, P., Ritschel, T., Vázquez, P.-P., Vinacua, À., Ropinski, T., 2018. Monte carlo convolution for learning on non-uniformly sampled point clouds. *ACM Transactions on Graphics (TOG)*, 37(6), 1–12.

Jiang, C. M., Huang, J., Kashinath, K., Prabhat, Marcus, P., Niessner, M., 2019. Spherical CNNs on unstructured grids. *International Conference on Learning Representations*.

Liu, K., Boehm, J., 2015. Classification of big point cloud data using cloud computing. *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40, 553–557.

Qi, C. R., Su, H., Mo, K., Guibas, L. J., 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 652–660.

Qi, C. R., Yi, L., Su, H., Guibas, L. J., 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*.

Rusu, R. B., Blodow, N., Beetz, M., 2009. Fast point feature histograms (fpfh) for 3d registration. *2009 IEEE international conference on robotics and automation*, IEEE, 3212–3217.

Salti, S., Tombari, F., Di Stefano, L., 2014. SHOT: Unique signatures of histograms for surface and texture description. *Computer Vision and Image Understanding*, 125, 251–264.

Tabkha, A., Hajji, R., Billen, R., Poux, F., 2019. Semantic enrichment of point cloud by automatic extraction and enhancement of 360° panoramas. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42(W17), 355–362.

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L. J., 2019. Kpconv: Flexible and deformable convolution for point clouds. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6411–6420.

Wang, Y., Shi, T., Yun, P., Tai, L., Liu, M., 2018. Pointseg: Real-time semantic segmentation based on 3d lidar point cloud. *arXiv preprint arXiv:1807.06288*.

Weinmann, M., Schmidt, A., Mallet, C., Hinz, S., Rottensteiner, F., Jutzi, B., 2015. Contextual classification of point cloud data by exploiting individual 3D neigbourhoods. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences II-3 (2015), Nr. W4*, 2(W4), 271–278.

Weisstein, E. W., 2018. Gnomonic projection. Publisher: Wolfram Research, Inc.

Wu, B., Wan, A., Yue, X., Keutzer, K., 2018. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 1887–1893.

Wu, W., Qi, Z., Fuxin, L., 2019. Pointconv: Deep convolutional networks on 3d point clouds. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9621–9630.

Xiao, T., Liu, Y., Zhou, B., Jiang, Y., Sun, J., 2018. Unified perceptual parsing for scene understanding. V. Ferrari, M. Hebert, C. Sminchisescu, Y. Weiss (eds), *Computer Vision – ECCV 2018*, Springer International Publishing, Cham, 432–448.

Yu, D., Ji, S., 2019. Grid Based Spherical CNN for Object Detection from Panoramic Images. 19(11), 2622. https://www.mdpi.com/1424-8220/19/11/2622.

Zhang, C., Liwicki, S., Smith, W., Cipolla, R., 2019. Orientation-aware semantic segmentation on icosahedron spheres. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 3532–3540.

Zhao, Q., Dai, F., Ma, Y., Wan, L., Zhang, J., Zhang, Y., 2018a. Spherical superpixel segmentation. *IEEE Transactions on Multimedia*, 20(6), 1406–1417.

Zhao, Q., Zhu, C., Dai, F., Ma, Y., Jin, G., Zhang, Y., 2018b. Distortion-aware CNNs for spherical images. *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 1198–1204.

Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A., 2017. Scene parsing through ade20k dataset. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 633–641.