3D Scene Semantization using 2D Image Segmentation Models: Application to Public Buildings

Sven Oesau, sven.oesau@cstb.fr
Centre Scientifique et Technique du Bâtiment, Sophia Antipolis, France

Mathieu Thorel, mathieu.thorel@cstb.fr
Centre Scientifique et Technique du Bâtiment, Sophia Antipolis, France

Abstract

3D object detection in point clouds is a recurring challenge. While image-based object detection shows satisfying results, the application of Deep Learning methods to point clouds is still challenging. The creation of training data in point clouds is time-consuming, but often necessary as industrial applications require specific labels. We propose a combined image and point cloud-based approach to localize installations in huge point clouds of public buildings. The Mask R-CNN model, trained on limited training data, is employed on reconstructed panoramic images to reduce the detection complexity to a limited number of potential occurrences. The exact position of objects is obtained by point registration of reference objects. The false positive detections can be eliminated by a geometric verification. The combined approach demonstrates the use of 2D machine learning techniques for a quick processing of huge data sets and the selective application of more expensive geometric verification for exact object localization.

Keywords: Point clouds, Object detection, Deep learning, Instance segmentation, Point cloud Registration, Hausdorff distance

1 Introduction

Building Information Modelling (BIM) is widely used in the Architecture, Construction and Engineering domain as well as for facility management due to its potential not just in the planning stage, but also for building life cycle management, renovation or physical simulations like HVAC. However, the buildings are often not exactly constructed according the CAD plans, have undocumented changes or never had digital plans to begin with. Thus, there is an increasing demand for creating an accurate as-built model of the existing building.

Nowadays, the creation of a BIM from an acquired point cloud is a manual or an assisted process requiring larger amounts of interaction. One of the major challenges is the lack of semantics in the acquired data. A sample may have been taken from a permanent structure, but also from installations, furniture, clutter like cloth or even humans and animals walking through the scanning process. Beyond the creation of a digital model of the permanent structures, many applications require a detection of installations and contained objects.

We propose a combined approach of image-based machine learning techniques and geometric point-based measures to detect objects with their exact position and orientation within large point clouds. While the application of machine learning methods for semantization is an obvious choice, the major challenge is the collection of training data. A mixed approach is proposed that provides satisfying results in the use case of detecting installed ticket dispensers and ticket control machines in point clouds of French train stations.

2 Related Work

Object detection in images has been a research topic since a long time but progressed quickly since the success of deep learning methods. Various detection tasks can be solved by training neural networks. Image classification methods constitute the basis of many more advanced deep learning methods by assigning a single or multiple labels to an image. Famous neural networks for this task are VGG (Simonyan & Zisserman 2014), GoogLeNet (Szegedy et al. 2015) and ResNet (He et al. 2016). Object detection methods localize objects inside the image and provide a labelled bounding box per object. They often integrate image classification methods for assigning a label to each subsection. The two most cited methods are Yolo (Redmon & Farhadi 2018) and Faster R-CNN (Ren et al. 2015). Mask R-CNN, an advancement of Faster R-CNN, was introduced by (He et al. 2017) and extends the method to instance segmentation. In addition to a bounding box, the silhouette of each object is provided as a pixel mask.

While point clouds are the method of choice for BIM modeling, due to their high accuracy and resolution, the state of the art of machine learning methods based on point clouds is still quite behind image-based methods. Compared to images, point clouds are unstructured and have a high variability in sampling density. In the last years, different approaches have been presented to apply neural networks to point clouds. Some methods have shown good results on aerial lidar (Blomley et al. 2016), e.g., random forest, but do not translate well into the indoor domain. SnapNet, introduced by (Boulch et al. 2018), generates images from different views in the point cloud and applies regular convolutional networks to perform a semantic segmentation, i.e., assigning a label to each pixel. The labels from the images are projected back onto the point cloud to obtain a semantized point cloud. Another approach of discretizing point clouds was followed by (Tchapmi et al. 2017). A 3D voxel grid is generated from the point cloud and a 3d convolutional neural network is applied for semantization. Promising results by using a graph gated neural network without discretization were presented by (Landrieu & Simonovsky 2018). They perform a clustering of points to reduce complexity and learn contextual relation between objects. They achieve a high precision, however, the processing time for medium sized data sets, ~80M points, is already quite high, ~2h. Point cloud-based methods are an active research field and new methods are presented each year. However, the effort to create training set for a specific task is a time-consuming manual task.

Another common approach of object detection in robotics is presented by (Aldoma et al. 2012). They compare various methods to locate keypoints in kinect data, a popular low-cost 3d camera introduced by Microsoft in 2009 and generate descriptors on given objects. Classical machine learning is used to identify known objects from the generated descriptors. While these approaches achieve good recognition rates, they are difficult to apply to large datasets of several hundred gigabytes.

3 Method overview

Out method provides the following contributions:

- **Combined image/point cloud-based approach for high precision**: We combine methods from the well-developed images-based object detection domain with point cloud-based methods to provide accurate position and orientation in 3D.
- **Requires 2D annotations instead of annotated point clouds**: The manual annotation of point clouds for an individual application is a time-consuming task. Our method uses

2D annotations in the form of silhouettes. Depending on the sought-after objects, even bounding boxes may be sufficient.

- **Promising results with smaller amounts of training data**: The two-step approach allows to compensate for the potentially lacking performance of the image-based object detection due to smaller amount of training data.

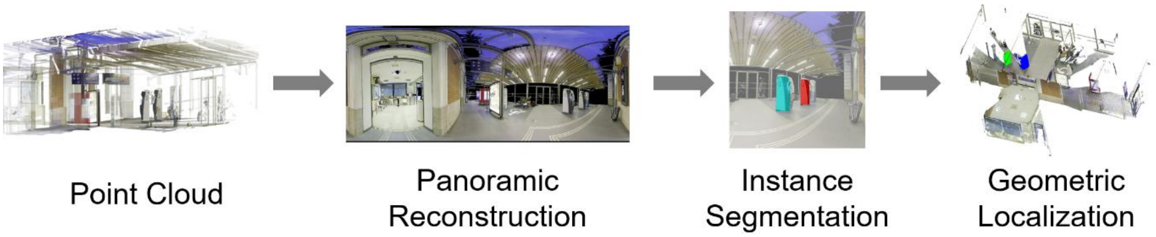

Figure 1. Pipeline of the presented method. Panoramic images are recreated from the input point cloud. A neural network performs an instance segmentation to detect the sought-after objects. The 3d positions and orientations are found after back projecting the detection mask into point cloud space. False positive detections are eliminated via geometric verification.

Lidar scanners can capture their surrounding with a very high resolution and produce millions of samples per second. Complete scans of public buildings, e.g., train stations, can thus contain several billions of points and have a file size of several hundred gigabytes. Processing that large amounts of data limits the choice of methods as global searches become computationally expensive.

Our method utilizes image-based object detection to narrow down the search complexity in large point clouds to a few occurrences, detailed in chapter 4. Often, images are taken during the acquisition process form the scanning positions to provide color information for each point sample. If no images are provided, a panoramic image is reconstructed from the colored points of each scanning position. The global search in the dataset is solved by applying a convolutional neural network for object detection onto the panoramic images.

The detection results, potential occurrences of the sought-after objects, undergo a geometric verification in 3d space to suppress false positive detections while retrieving the exact 3d position and orientation, detailed in chapter 5. Using the reconstructed panoramic images from scanning positions for object detection, allows to project the image masks into the point cloud to obtain an estimated 3d position of the detection, as the positions and orientations of scanning positions are known from the registration process. Registering a point cloud of the reference object, which may be extracted manually from the point cloud once or sampled from a CAD representation, with the full acquired point cloud at the estimated 3d position provides the exact position and orientation of the detection. Finally, the geometric verification helps differing correct from false detections, by considering the registration quality and the geometric differences between the reference object and the acquired point cloud.

For some applications, it is difficult to collect sufficient training data, e.g., if the sought-after objects are particular or the acquisition method produces distorted images, e.g., panoramic images. The presented method can, however, compensate for a lacking performance due to limited training data. Objects are often seen in more than just one scan, and a single detection is sufficient to identify the object in the point cloud.

4 Object detection in panoramic images

The first part of the method operates on images and helps to narrow down the object detection to a smaller number of occurrences. The panoramic images converted from colored point samples are fed into a Mask R-CNN network for object detection. The precision of this step may be lacking due to a low amount of training data. This is compensated in the subsequent geometric step, see chapter 5.

4.1 Conversion of single scans into panoramic images

Although most laser scanners capture images to provide color information for each point sample, not always are those images distributed with the point data. As LiDAR scanners usually have a low angular distortion, it is possible to reconstruct a panoramic image from a colored pointset with a known position and vertical direction. The coordinates of the panoramic image correspond to the azimuth ϕ and elevation θ angles of each point. To convert the points, they have to be transformed into the space of the corresponding scanning position. The two angles can be calculated from the normalized vector (x, y, z):

$$\theta = arccos(z)$$

$$\phi = \begin{cases} \frac{\pi}{2} - \tan^{-1}\left(\frac{x}{y}\right), \text{ y>0} \\ \pi + \tan^{-1}\left(\frac{x}{y}\right), y < 0 \end{cases}$$

A converted image is shown in Figure 2. The scanner only generates samples, where it captures a reflected laser signal. The uncovered parts of the image are shown in black.

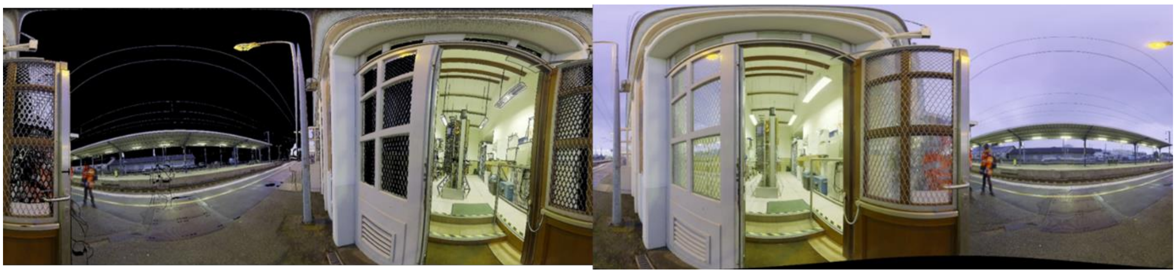

Figure 2. Left: Converted panoramic image from color point cloud. No points have been collected in the sky, as there is no surface. Some artifacts have been caused by people moving through the scene during the scanning process. **Right:** Actual photo taken be the Lidar scanner. A panoramic photo is periodic, the periodic shift results from the point cloud registration during the acquisition process.

4.2 Mask R-CNN for object segmentation.

The reconstructed panoramic images provide a direct link between the pixels in the image and the point samples in the acquired point cloud. Thus, a detection in the image can be converted into a set of points. Various deep learning methods exist to perform this task. Mask R-CNN is particularly suitable as a state-of-the-art instance segmentation method as it provides a mask which assigns a set of pixels to a detected object. However, depending on the sought-after object types, a regular object detection method providing only a bounding box instead of a mask may be sufficient, especially for large compact objects whose silhouette is close to a rectangular shape than for thin elongated shapes.

Training Mask R-CNN on a smaller set of panoramic images comes with two challenges. Complex models trained on a limited training set often leads to an overfitting of the model. Overfitting describes an adaption of the model to noise and single individual examples and is a common problem for complex models. It results in high precision on the used training data, but a significant lower precision on new data. This method heavily relies on data augmentation to increase the amount of available training data. Data augmentation generates new training data from existing samples, by applying simple modifications. Those modifications introduce alterations to the image, which does not impact the presence of the object. E.g., adding noise or blur to an image, mirroring or rotating the image or using crops of the original image which still contain the silhouette of the labeled object.

Another established technique that can be employed against overfitting is dropout which was introduced by (Srivastava et al. 2014). Dropout randomly nodes in the neural network during the training process. While not being very intuitive, it has become a well-established technique in the community.

The second challenge is the large size of panoramic images and their strong distortion, see Figure 2. Depending on the used LiDAR scanner and chosen resolution, the number of points in single scans, and thus the corresponding panoramic images, can exceed 40 million. Many neural networks have specific image input size or do not scale well with large image sizes. To not lose information, the input images are cut into several smaller images instead of a downscaling.

The distortion in the panoramic images results from the conversion of the point cloud generated through a spherical scanning process to a rectangular image. Thus, the distortion is the highest at the lower and upper boundary of the image and the distortion is small in the horizontal middle part of the image. In the use case of detecting installations for direct human interaction, the most likely area in the image falls into the area of lowest distortion.

Thus, the input images are cut onto 1024x1024 px subimages centered along the horizontal line. The remaining upper and lower parts of the image are equally cut into subimages, which may overlap as the dimensions of input images typically are not a multiple of 1024.

5 Geometric object localization

Starting from the masks generated by the Mask R-CNN the estimated 3d position of the potentially detected object is calculated. As almost all pixels in the panoramic image correspond to a 3d point, the estimated 3d position is given by the average position of the points covered by the mask. Pixels in the panoramic image that are not covered by points, e.g., the sky in Figure 2, are simply excluded. Starting from the estimated 3d position the point cloud of the reference object is locally registered with the acquired point cloud to obtain an exact orientation and position in 3d. In case of false positive detections this step is already likely to fail as there are not sufficient local correspondences between the geometry of the reference object and the acquired point cloud. After aligning the reference point cloud, the point clouds are compared to verify identical geometry.

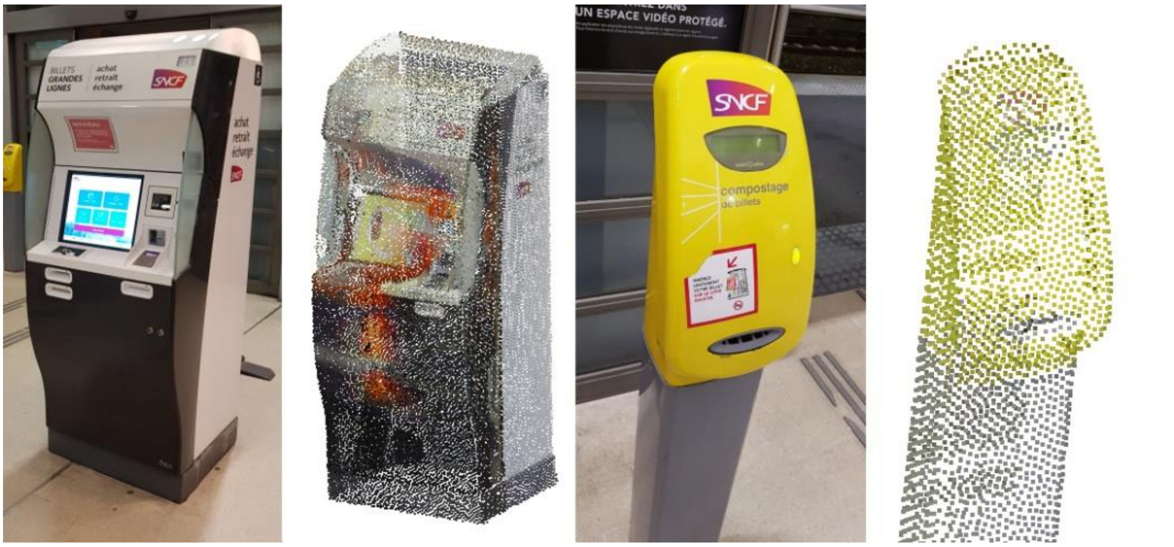

Figure 3. Left: Photograph of a “main line” ticket dispenser and the reference point cloud. **Right:** A photograph of the second object, a ticket control machine and its reference point cloud.

5.1 Reference objects

In this paper, the points cloud segmentation use case is an inventory of two specific furniture equipment in train stations: *"main line" ticket dispensers* and *ticket control machines*. These specific objects, designed from the same factory, allow to restrict the detection to their exact geometry. Reference objects, i.e., a small point cloud, of the sought-after equipment are used in the geometric step for detecting the exact position and orientation, as well as to perform a geometric verification to reject false positive detections from the machine learning part on similar objects. The reference object can be manually extracted from a point cloud or automatically generated from a CAD model by sampling the outer envelope. The resolution of the reference point cloud is down sampled to at most 1 point per cm^3. Some missing parts in the reference objects are tolerable, e.g., the bottom and top surfaces as they are hidden or not visible from the scanner position. The reference point clouds of the sought-after objects in the use case of train stations are shown in Figure 3.

5.2 3D alignment

The reference point cloud is locally aligned with the predicted object in the acquired point cloud via *point cloud registration method* following a similar approach as presented by (Aldoma et al. 2012). Point cloud registration can be divided into two different categories: global and local registration. Local registration, e.g., Iterative Closest Point, requires the two point clouds to be roughly aligned and iteratively aligns the two point clouds by minimizing the distance. The estimated position resulting from the Mask R-CNN detection generally does not provide a sufficient close alignment. Thus, a global registration is required which instead of directly minimizing the distance between the two point clouds aims at detecting keypoints in both point clouds. Keypoints are distinctive points with a specific neighborhood. This method uses the Scale-Invariant Feature Transform (SIFT) keypoint of the Point Cloud Library (Rusu & Cousins 2011), a 3D variant of the original SIFT keypoint for images introduced by (Lowe 2004).

Instead of calculating the keypoints for the whole acquired point cloud, which is very expensive considering the common size of several hundred gigabytes of point data, only a local area of the acquired point cloud with a radius of 1m around the estimated 3d position is considered.

Aligning the two point clouds is solved by finding correspondences for the keypoints of the small reference point cloud within the keypoints of the acquired point cloud and minimizing the distance between them. As this is an error prone process, the RANSAC principle (Fischler & Bolles 1981) is applied that aligns a random small number of corresponding keypoints and uses the remaining keypoints to verify a good alignment. This process is repeated to find an alignment that fits the largest number of keypoints. After the keypoint-based global registration, a local registration using *Iterative Closest Point* is applied for a fine alignment. The result of registering a reference point cloud with the acquired point cloud is shown in Figure 4.

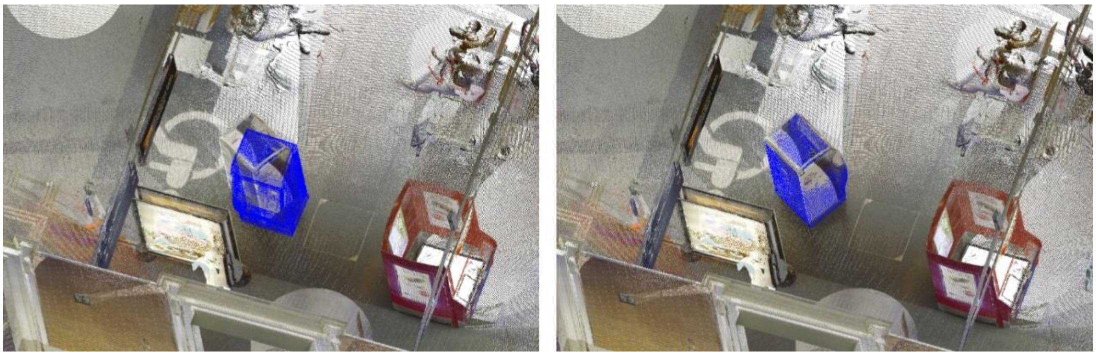

Figure 4. Left: Reference object placed at the estimated 3d position before registration. **Right:** After registering the reference object is well aligned with the real object.

5.3 Geometric Verification

After aligning the reference model with the acquired point cloud, the aligned geometry of the reference object should match the acquired point cloud in case of a correct detection. The point cloud registration, however, will always provide the best match as a result, no matter if those two point clouds actually share a similar geometry or not. As the performance of the machine learning method strongly depends on the amount of training data, the number of false positives may be high if not much training data is available. Thus, after registration a comparison between the aligned reference object and the acquired point cloud is performed to reject a false detection. To validate a correct detection the point cloud of the reference object is compared in its aligned position within the acquired point cloud.

The geometry is compared by checking if in the acquired point cloud there are points close to the points of the reference object. In other words, it is verified that there is an object in the scene that has the same shape as the reference object.

For verifying the geometry of the object, the one-sided Hausdorff-distance is used, i.e., for each point of the reference object, the distance of the closest point in the acquired point cloud is calculated, see Figure 5. If the distance is below 3cm, the geometric matching is assumed correct. In addition, the orientation of the aligned reference object is considered. If the alignment is non vertical, the detection is discarded.

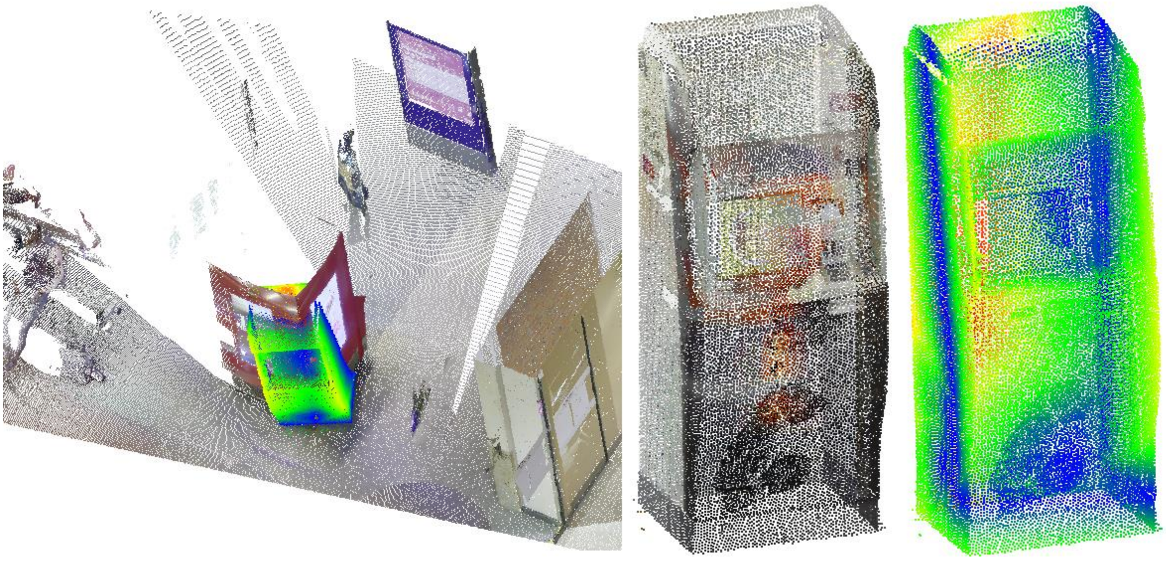

Figure 5. Left: False positive detection of a ticket dispenser on a soda vending machine. The registration does not provide a proper alignment as the objects do not match. **Right:** Original point cloud of reference object and the same point cloud colored by the distance of the closest point in the acquired point cloud. Blue color indicates a satisfying close alignment and red indicates a distance of more than 20cm.

6 Experiments

The method has been implemented by using the public Mask R-CNN implemented provided by Matterport (Matterport 2019) as well as the Point Cloud Library (Rusu & Cousins 2011) for the point cloud registration and distance calculation. The manual labeling of the training dataset for our use case is done with the COCO Annotator tool (Brooks 2019).

The point clouds of 5 different train stations were used in this study. The smallest one consisting of 33 scanning positions and around 390M points and the largest one with 576 scanning positions and approximately 18 billion points. 126 different pictures containing one of the two reference objects, see Figure 3, have been identified and labeled in order to build the machine learning 2d dataset. However, in many cases the objects are quite small in the background.

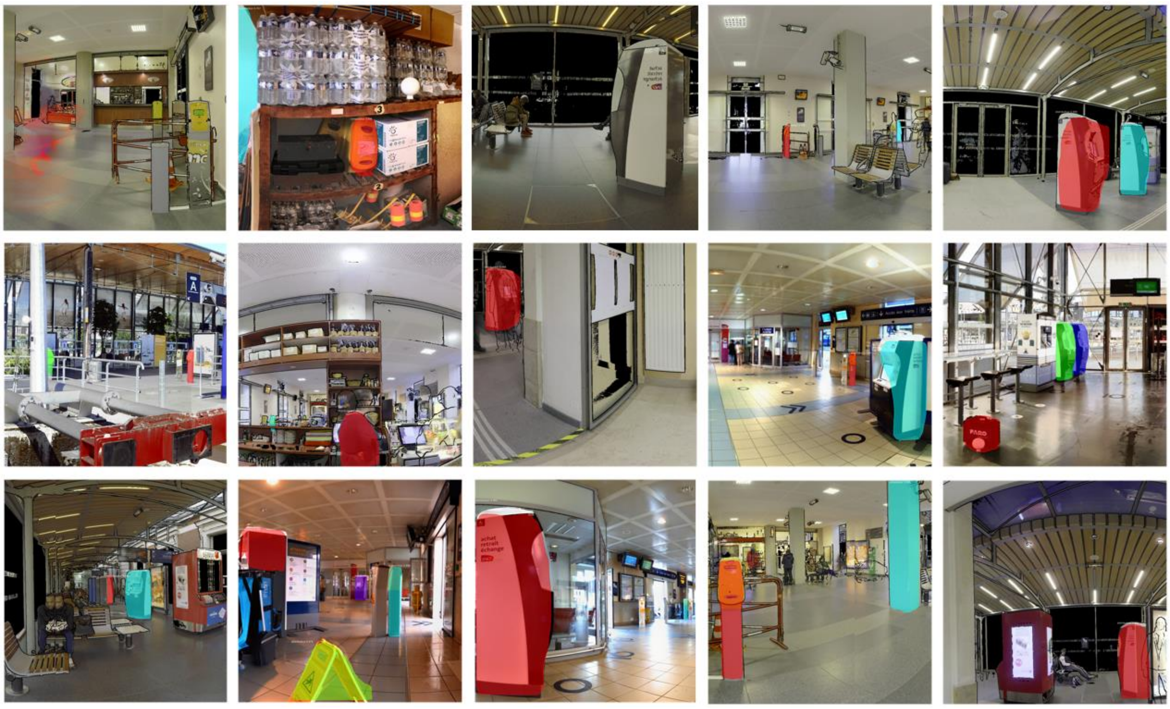

Figure 6: Results of the Mask R-CNN step before the geometric localization step to suppress false positive detections. Trained on crops from 111 reconstructed panoramic images, the results of the classifier on 15 images are shown above. Out of a total of 27 occurrences, 22 have been detected correctly. There are 6 false positives and 5 false negatives, i.e., missed occurrences. The subsequent geometric verification achieved a suppression of all false positives. In most cases the missed occurrences are in the background with one major exceptions, shown in the mid images in the top row.

The training of the Mask R-CNN neural network used the pretrained network from the COCO dataset (Lin et al. 2014) to speed up the training. For comparison of the performance with low amounts of training data, two models have been trained. One on the limited number of 111 annotated panoramic images and one on a training set of 466 images including photos manually taken by the authors using regular cameras and images collected from the web. The results of the model trained only on panoramic images is shown in Figure 6. Out of 27 occurrences of the sought-after objects, 22 were detected correctly, 5 objects were missed and additional 6 false positives were reported. The geometric verification successfully suppressed all 6 false positives leading to a detection precision of 1,0 with a recall of 0,815.

The second model trained on a much larger training set using regular photos achieved a detection performance of 23 detected objects, 4 missed occurrences and 5 false positives. After successful suppression of the false positives by the geometric verification, a precision of 1,0 and a recall of 0,851 was achieved. Both evaluations did not consider, that some pictures showed the same actual instance of an object in different images. In this case, a single correction is sufficient.

However, due to the limited number of object occurrences in the reconstructed panoramic images into account, multiple occurrences of the same object were not considered in the evaluation. Using a large number of regular images showed to improve the detection performance compared to the much smaller number of reconstructed panoramic images. Considering that the same instance of an object typically appears in 3 to 4 images according to the provided datasets both models provide a satisfying detection performance.

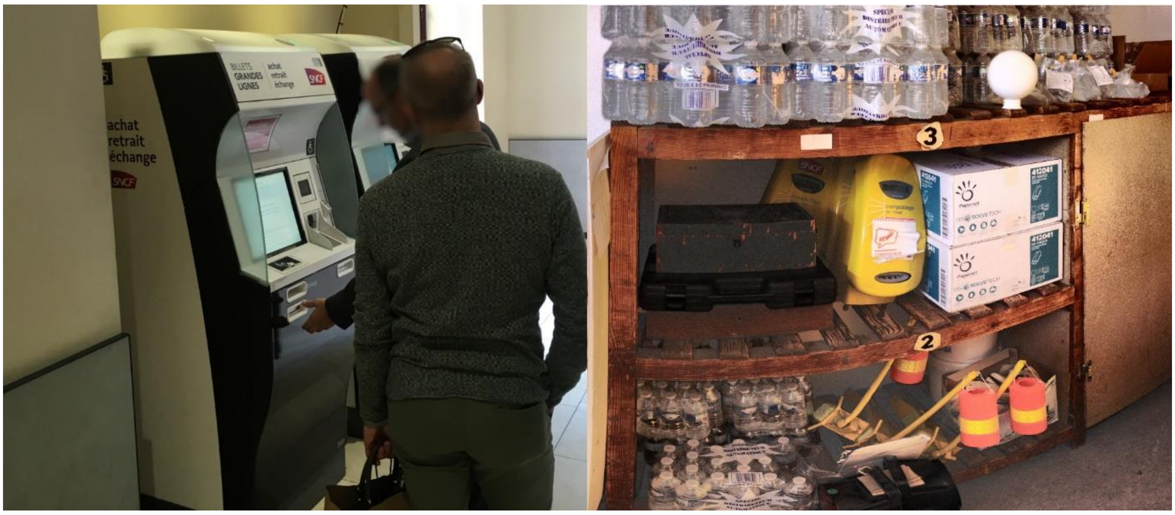

Figure 7: Challenging cases for the method. **Left:** Larger parts of ticket dispenser may be hidden in the wall and additionally occluded by people. Due to operational reasons, the acquisition in public buildings often happens during operational hours. **Right:** Although the AI based object detection correctly identified the ticket control machine, the geometric verification rejected those detections due to non-vertical orientation. It can be discussed if the object should be detected in this case or not.

6.1 Limitations

The main limitation of our method is the requirement of reference objects. However, providing an orientation of detected objects is difficult without knowing the shape of the object. Additional limitations come from the strict geometric verification. Although the *ticket control machine* in the storage room, see Figure 7, were detected by the machine learning method, the detection was later rejected.

6.2 Conclusion

This paper presents an instance segmentation method for unstructured point cloud scenes. Due to the lack of existing point cloud datasets with suitable labels and the current complexity of handling huge volumes of points, the authors chose to mix well known image segmentation machine learning technics (Mask R-CNN approach) with point cloud registration methods. Obtained results are a partially labelled point cloud scene. The covered use case brings some methodological simplifications: target objects need to have a fixed geometry (industrial manufacturing), allowing the use of reference point cloud objects. A dataset composed of 466 images is created and manually labelled in order to train a Mask R-CNN model. This model is applied to images extracted from colorized point cloud scenes. Outputs are 2D instance class masks that are mapped on the raw point cloud. Then, point cloud registration techniques (SIFT, Iterative Closest Point, Hausdorff-distance) are applied to refine predictions into the point cloud scene.

The presented approach deals with limited amounts of training data for machine learning in industrial applications while still achieving promising results. If very large amount of training data was available at some point, it could be possible to include the orientation of objects into the learning process and finally avoid the need of reference objects. The orientation of an object could be learned as an additional label, e.g., “ticket dispenser machine” & “facing_angle_45_deg” providing the relative orientation of the object towards the scanner. However, removal of the geometric localization would also introduce a lower precision in 3d position and probably orientation.

The proposed segmentation method is suitable for detection and tag of industrial objects having a fixed geometry. Future works will be focused on parametric objects such as doors and windows. Last types of objects, the more difficult, are those with non-predictable geometries such as floors.

6.3 Acknowledgements

This work was funded by SNCF Gares & Connexions, part of the French railway company. We thank them additionally for their input and for providing us large amounts of data.

References

Aldoma, A., Marton, Z., Tombari, F., Wohlkinger, W., Potthast, C., Zeisl, B., Bogdan Rusu, R., Gedikli, S. & Vincze, M. (2012). Three-Dimensional Object Recognition and 6 DoF Pose Estimation. IEEE Robotics & Automation Magazine. pp. 80-91. https://ieeexplore.ieee.org/document/6299166

Blomley, R., Jutzi, B., Weinmann, M. (2016). Classification of airborne laser scanning data using geometric multi-scale features and different neighbourhood types. *ISPRS annals III-3.* https://doi.org/10.5194/isprs-annals-III-3-169-2016

Boulch, A., Guerry, J., Le Saux, B., Audebert, N. (2018). SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks. *Computers & Graphics* 71. https://doi.org/10.1016/j.cag.2017.11.010

Brooks, J. (2019). COCO Annotator. https://github.com/jsbroks/coco-annotator/

Fischler, M. and Bolles, R. (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM 24 (6)*. https://doi.org/10.1145/358669.358692

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *Conference on Computer Vision and Pattern Recognition.* https://doi.org/10.1109/CVPR.2016.90

He, K., Gkioxari, G., Dollár, P. & Girshick, R. (2017). Mask R-CNN. International Conference on Computer Vision. https://doi.org/10.1109/ICCV.2017.322

Landrieu, L. & Simonovsky, M. (2018). Large-Scale Point Cloud Semantic Segmentation with Superpoint Graphs. *Conference on Computer Vision and Pattern Recognition.* https://doi.org/10.1109/CVPR.2018.00479

Yi Lin, T., Maire, M., Belongie, S., Bourdev, L., Girshick , R., Hays, J., Perona, P., Ramanan, D., Dollár, P. & Zitnick, L. (2014). Microsoft COCO: Common Objects in Context. http://arxiv.org/abs/1405.0312

Lowe, D.G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision 60*. https://doi.org/10.1023/B:VISI.0000029664.99615.94

Matterport (2019), Mask R-CNN for Object Detection and Segmentation, https://github.com/matterport/Mask_RCNN

Redmon, J., Farhadi, Al. (2018). YOLOv3: An Incremental Improvement. *http://arxiv.org/abs/1804.02767*

Ren, S., He, K. Girshick, R. & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (6).* https://doi.org/10.1109/TPAMI.2016.2577031

Rusu, R. & Cousins, S. (2011). Point Cloud Library. *IEEE International Conference on Robotics and Automation. http://pointclouds.org*

Simonyan, K. & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *International Conference on Learning Representations.* http://arxiv.org/abs/1409.1556

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research 15 (1).* https://dl.acm.org/doi/10.5555/2627435.2670313

Szegedy C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A. (2015). Going Deeper with Convolutions. *Conference on Computer Vision and Pattern Recognition.* https://doi.org/10.1109/CVPR.2015.7298594

Tchapmi, L. P., Choy, C. B., Armeni, I., Gwak, J. & Savarese, S. (2017). SEGCloud: Semantic Segmentation of 3D Point Clouds. *2017 International Conference on 3D Vision (3DV).* https://doi.org/10.1109/3DV.2017.00067