EUROGRAPHICS Workshop on Graphics and Cultural Heritage (2023)
A. Bucciero, B. Fanini, H. Graf, S. Pescarin, and S. Rizvic (Editors)

Short Paper

Automatic detection of windows reflection or transparency pollution in TLS acquisitions

E. Badalyan[1], A. Schenkel[1] and O. Debeir[1]

[1]Laboratory of Image Synthesis and Analysis, Ecole polytechnique de Bruxelles, Université Libre de Bruxelles, Belgium

Abstract

Three-dimensional acquisitions have been more and more used in recent years, for multiple applications, such as cultural heritage preservation. When these point clouds are generated through laser scanning, transparent and/or reflective objects such as windows can generate inexact or undesirable data. These must be cleaned up by a human, which is often time-consuming and requires experience. This work provides an insight on some methods that can be used to automate this task. It investigates the usage of Mask R-CNN with intensity images in equirectangular projections. The huge images are tiled into squares of 2048x2048 pixels for both training and prediction. The model has good performances on the test and validation sets to handle both types of problems; but also to manage the presence of a mirror in a scene.

CCS Concepts

• ***Computing methodologies*** → ***Object detection; Image segmentation;***

1. Introduction

Three-dimensional acquisitions have seen a large increase in usage in recent years: from virtual reality [TBP16] to cultural heritage preservation [PPM*20], the applications are widespread. One of the main uses of point clouds is to generate scans of buildings, either from the inside or the outside. Buildings can be scanned for research purposes, but also to create the possibility of organizing virtual visits, and even to have precise models of buildings for world heritage or for their study (plan production, modification, simulation, etc.).

1.1. Problem

Reconstructions of building models, whether acquired by laser scanning or by photogrammetry, can be misformed due to reflective or transparent objects. The presence of windows is particularly problematic during data acquisition. Indeed, depending on several factors such as the condition of the windows' glasses, the relative position of the measuring device, the luminosity on both sides, the windows are either transparent or reflective. A same acquisition can even present these two cases simultaneously, as illustrated in Figure 1. Transparency introduces the presence of points which may not be desired, and may have distortions due to refraction. Reflection is also an issue, since cameras or scanners - obviously not knowing that the ray has been reflected - would place points, representing the environment, in incorrect positions.

In both cases, the risk is to produce a set of points, potentially coherent, behind the problematic object. This undesirable data must indeed be filtered by a human, which is often time-consuming and requires experience. This paper's contribution is thus a method to automate this task. It will be focused on the cleaning of data acquired with terrestrial laser scanners, taking into account the following choices:

- Equirectangular image: individual scans are ordered in a matrix shape, like an image. The image format has the advantage of having a wide scientific literature regarding its analysis and segmentation approaches compared to a point cloud format, and generally requires fewer resources for its manipulation;
- Intensity-based data: common TLS provides the position of each point as well as an uncalibrated intensity data (laser beam backscattered reflectance). Color information can supplement measurements; but requires an additional sensor that can take configurations depending on the device (on-board camera, off-center DSLR or 360 camera) and regularly presents problems (i.e. over or underexposure, calibration errors);
- Individual scan treatment: merging data into a single model can lead to the interleaving of valid and erroneous values, making it difficult to filter them, even manually.

2. State of the art

Many Deep Learning methods were proposed in image classification and segmentation and outperform the classical methods [MBP*21]. Among these, Mask R-CNN [HGDG17] is one of

DOI: 10.2312/gch.20231163

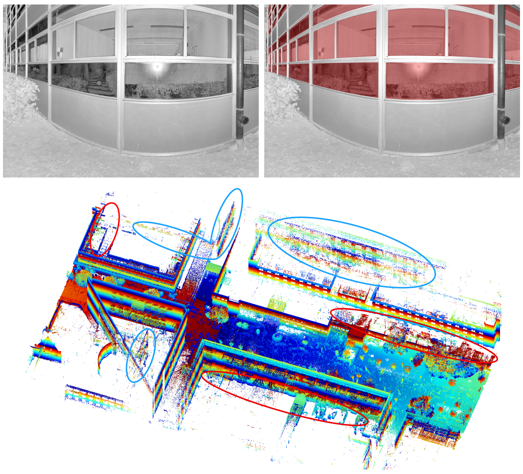

Figure 1: *Examples showing both transparency and reflection. Top left: on a 2D representation; the upper row of windows shows points inside the building, while the lower row reflects the bushes and the white sphere. Top right: the corresponding detection. Bottom: on a 3D representation with color used to encode elevation; the points circled in blue are caused by reflection, while those circled in red are due to transparency.*

the most popular ones and usually produces good results [BP20, MBP*21].

Karara et al. [KHP21] use mask R-CNN to perform instance segmentation of everyday objects in unordered point clouds, transformed in images using spherical, cylindrical, and cubic projections. The results are projected back onto the point cloud. Nordmark et al. [NA21] base their approach on mask R-CNN and transfer learning to recognize windows on RGB camera images. However, their approach yielded relatively weak results when applied to intensity images. Tan et al. [TLCS21] work RGBD datasets with Mask R-CNN to detect mirror and PlaneRCNN to correct them, only dealing with part of our problem with other kind of data.

There also has been some research on direct segmentation of point clouds [NK18], and a few methods have been proposed, such as PointNet [QSMG17] and VoxelNet [ZT18]. Image segmentation is chosen over these methods due to the complexity of the task (i.e. cleaning of all problematic points) and the large dataset size, which would result in lengthy training times.

The key distinction between the aforementioned works and our task lies in the availability of color information. Color plays a crucial role in object recognition for both humans and computer vision algorithms. However, our dataset solely provides intensity information, setting it apart from the others. Additionally, while Karara et al. [KHP21] aimed to segment common everyday objects according to the Microsoft COCO dataset [LMB*14], our focus is on segmenting windows.

Notably, Nordmark et al. [NA21] did not specifically aim for precise point segmentation, allowing for more lenient annotations that encompassed the entire window frame as rectangles. Conversely, our approach strives for utmost precision by solely capturing the glass portion of the windows. This, combined with image deformations, results in annotations of varying shapes, which could potentially pose challenges for the model to learn accurate window segmentation.

In summary, existing research has not presented a solution for effectively segmenting challenging objects like windows in images obtained from extensive point clouds, especially when color information is absent. However, this work demonstrates that Mask R-CNN, coupled with transfer learning on new intensity-only data, can successfully segment novel object types.

3. Materials and Methods

3.1. Data

The dataset is made up of 98 individual raw scans (no cleaning filter has been previously applied to the data), each taken from outside a building, and providing positions and intensities of the points. There are mainly two different resolutions of scans in the dataset: some contain about 40 million points, while the others have more or less 160 million. Out of the 98 scans, 78 are in the training set, while the test and validation sets have 10 each.

3.2. Annotations

For windows that are closer to the foreground, only the glass is taken, hence if the frame has a lattice separating the window in multiple pieces of glass, then each is annotated separately. However, for windows that are farther in the background, the different glasses are not annotated separately. Since these windows represent only a few pixels in the image, regrouping them should not impact the model's accuracy.

3.3. Methods

Our approach is based on an image segmentation model applied on scan data transformed into a grayscale equirectangular image. These images are quite large and memory errors can occur during training. To avoid this, images are randomly cropped to 2048x2048 pixels instead of using the whole image, so that all regions of the image are processed by the model following the number of iterations.

This size problem is also present during prediction. However, instead of taking a single random crop, the image is tiled like a grid, and the tiles are fed to the model separately. Then, the tiled masks are stitched back together to form a single mask for the entire image.

There are sometimes border effects where the tiles meet, but they can be removed by a small amount of overlap, as shown in Figure 2. A minimal overlap of 50 pixels was found to be quite effective. For pixels that belong to multiple tiles, a logical OR of the multiple predictions is taken. Once the mask is obtained for the whole image, it is simply used to remove segmented points from the point cloud.

The Mask R-CNN model, and more precisely the mask_rcnn_R_50_FPN_3x [WKM*19] architecture is used, and transfer learning is applied by using weights that were pre-trained on the Microsoft COCO dataset [LMB*14]. The architecture with a depth of 50 is used because it trains much faster than the 101 one, with only a small loss in performance [HGDG17]. The model is trained over 10000 iterations, with a learning rate that starts at 0.001 and is halved at iterations [3000, 5000, 6000, 7000, 8000, 9000]. The first two blocks - out of five - of the ResNet are frozen, i.e. not trained because they are low-level abstraction layers, and thus their previous values should not change much. The loss function is the sum of the five following losses: RPN classification loss, RPN box regression loss, ROI heads classification loss, ROI heads box regression loss, and mask loss.

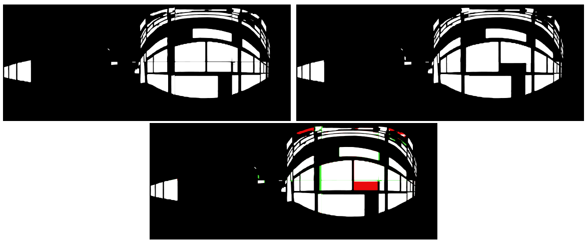

Figure 2: *Predicted mask with no overlap between the tiles (left) and predicted mask with 50 px of overlap (right), both with the model trained for 8000 iterations. As can be seen, the discontinuity in the segmented regions disappears. Bottom: The difference which the following color coding: black: in neither mask, white: in both masks, red: in the first mask but not the second, green: in the second mask but not the first.*

3.4. Augmentations

Different lighting conditions can largely affect the intensities of pixels. Thus, two intensity augmentations are applied to the training images to simulate this effect. A brightness augmentation will randomly choose a number $w_b \in [0.5, 1.5]$, multiply the value of each pixel by this number, then clip the result between 0 and 255. Furthermore, contrast will be manipulated by choosing another random number $w_c \in [0.5, 1.5]$ and then performing $o = (1 - w_c) * m + w_c * i$ where o is the output (augmented) image, i is the input image, and m is the average intensity of the input image [WKM*19].

4. Results

4.1. Evaluation on test and validation sets

The evaluation results of this model on the test set are shown in Figure 3, as a function of the number of iterations for which the model was trained. First of all, one can notice that the best F1-score is reached by the model after training for 8000 iterations, which might mean that 10000 iterations is a bit excessive and could lead to overfitting. This seems to be supported by the fact the model already has decent scores after 2000 iterations. An example of a prediction by this model is shown in Figure 4.

Furthermore, for all cases, precision is higher than recall which means that the model has less false positives than false negatives. It could be argued that false positives are more acceptable than false negatives for this use case, since the clouds are going to be merged, which means that points may appear multiple times. Thus, even if a point is incorrectly deleted from a scan, there is a possibility that the data will be kept in another one. If this point of view is taken, then maximizing recall may become a priority (the model trained for 4000 iterations would thus be chosen).

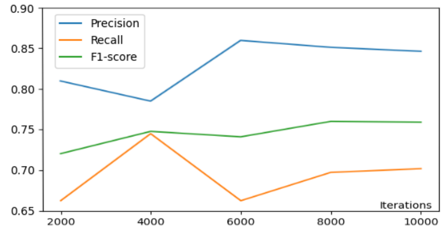

Figure 3: *Average Precision, Recall, and F1-score of the model on the test set as a function of the number of iterations for which it was trained*

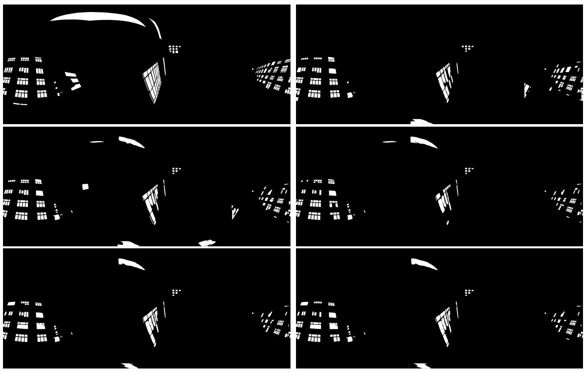

Figure 4: *Ground truth mask for one of the images of the test set (top left) and the mask predicted by the model trained for 2000 (top right), 4000 (mid left), 6000 (mid right), 8000 (bottom left), and 10000 (bottom right) iterations.*

With a best F1-score of 0.7601, the model seems quite efficient at finding windows. On the validation set, the average precision, recall, and F1-score are 0.8730, 0.7731, and 0.8157 respectively. Using the overlap raises the average F1-score on the test set to 0.7630 and on the validation set to 0.8228. The increase in scores is not that large, but the removal of the border effect is worth the change.

4.2. Evaluation on other cases

The model is capable of correctly segmenting some challenging images. Indeed, it can be seen in Figure 5 that it segments mirrors, which have a texture similar to windows, but were completely absent from the training dataset, which shows some robustness. This is particularly interesting since it is difficult to identify these problematic points that require careful inspection or deep knowledge of the digitized places.

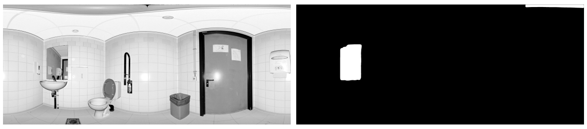

Figure 5: *Challenging image with a mirror (left) and the mask predicted by the model trained for 8000 iterations (right).*

4.3. Results on point clouds

The model is also tested by reprojecting the prediction masks onto a point cloud. Figure 6 shows the inner side of a building that was scanned from the outside. Due to the transparent windows, many points have appeared inside the building. In the cloud generated by reprojecting the hand-annotated mask, most of these points have been removed. The version generated using the mask predicted by the model also exhibits significantly less incorrect points, showing the quality of the model.

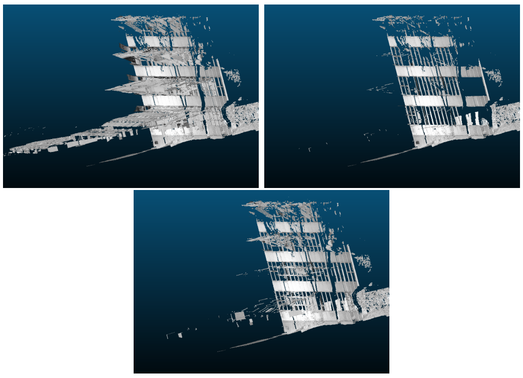

Figure 6: *Test of a prediction on a point cloud: raw (top left), cleaned with hand-annotated mask (top right), and cleaned by the model (bottom).*

4.4. Conclusion

The purpose of this work was to develop an automatic method to remove undesirable points introduced by windows and other reflective and/or transparent objects from point cloud data. Indeed, when laser scanners are used to produce point clouds, such objects can induce the presence of misplaced and unwanted points. The problem is solved by segmenting the windows out of images generated from the clouds, then using the resulting masks to clean the point cloud data. The model used to do this is Mask R-CNN, an image semantic and instance segmentation model that has been known to perform quite well. The creation of the image dataset was quite simple, as it simply required to put the intensities recorded by the laser scanner in an image in the order they were given in.

As for the predictions, the images were too large to be fed into the models as is. Hence, the images are divided into tiles that are segmented separately. Through various tests, it was shown that this method yielded reasonably good results, and it was demonstrated that this model could handle some challenging cases it had never encountered during training, such as other reflective surfaces. When reprojecting the prediction masks on the points clouds, a large part of the erroneous points were removed.

4.5. Future Work

First of all, enlarging the dataset to add variety, including scans from inside buildings, could lead to a significant increase of performance. Moreover, adding geometric information, such as depths or normals, could prove interesting. Using cubemap projections instead of the equirectangular images was also tested, but led to less satisfying segmentations. Investigating the reasons for these results and attempting to improve them could be the subject of future work. Finally, the problem of smaller windows being visible through bigger ones might be solved by training two separate models for big and small windows.

References

[BP20] Bharati P., Pramanik A.: Deep learning techniques—r-cnn to mask r-cnn: a survey. *Computational Intelligence in Pattern Recognition* (2020), 657–668. 2

[HGDG17] He K., Gkioxari G., Dollár P., Girshick R.: Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (2017), pp. 2961–2969. 1, 3

[KHP21] Karara G., Hajji R., Poux F.: 3d point cloud semantic augmentation: Instance segmentation of 360 panoramas by deep learning techniques. *Remote Sensing 13*, 18 (2021), 3647. 2

[LMB*14] Lin T.-Y., Maire M., Belongie S., Hays J., Perona P., Ramanan D., Dollár P., Zitnick C. L.: Microsoft coco: Common objects in context. In *European conference on computer vision* (2014), Springer, pp. 740–755. 2, 3

[MBP*21] Minaee S., Boykov Y. Y., Porikli F., Plaza A. J., Kehtarnavaz N., Terzopoulos D.: Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence* (2021). 1, 2

[NA21] Nordmark N., Ayenew M.: Window detection in facade imagery: A deep learning approach using mask r-cnn. *arXiv preprint arXiv:2107.10006* (2021). 2

[NK18] Neuhausen M., König M.: Automatic window detection in facade images. *Automation in Construction 96* (2018), 527–539. 2

[PPM*20] Pierdicca R., Paolanti M., Matrone F., Martini M., Morbidoni C., Malinverni E. S., Frontoni E., Lingua A. M.: Point cloud semantic segmentation using a deep learning framework for cultural heritage. *Remote Sensing 12*, 6 (2020). doi:10.3390/rs12061005. 1

[QSMG17] Qi C. R., Su H., Mo K., Guibas L. J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2017), pp. 652–660. 2

[TBP16] Tredinnick R., Broecker M., Ponto K.: Progressive feedback point cloud rendering for virtual reality display. In *2016 IEEE Virtual Reality (VR)* (2016), pp. 301–302. 1

[TLCS21] Tan J., Lin W., Chang A. X., Savva M.: Mirror3d: Depth refinement for mirror surfaces. *CoRR abs/2106.06629* (2021). 2

[WKM*19] Wu Y., Kirillov A., Massa F., Lo W.-Y., Girshick R.: Detectron2. https://github.com/facebookresearch/detectron2, 2019. 2, 3

[ZT18] Zhou Y., Tuzel O.: Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2018). 2