

On Voxel-Based Sampling of Planar Surface Areas from Photogrammetric UAV Data

Petri Rönholm¹, Matti T. Vaaja¹, Hannu Hyyppä¹

¹Department of Built Environment, Aalto University, Finland

Keywords: Planar surface areas, voxel, photogrammetry, UAV, 3D point cloud

Abstract

Photogrammetric methods, when used with UAV-based image acquisition, allow the creation of dense 3D point clouds that can be utilized in numerous applications across various fields. Planar surface areas can be extracted from 3D point clouds and utilized for various purposes, such as building reconstruction, augmented reality, general scene understanding, robot navigation, finding corresponding observations needed for data registration, or in SLAM, to name a few. The aim of this paper was to examine the behavior of voxel-based detection and sampling of planar surface areas from UAV-based photogrammetric 3D point clouds with both non-overlapping and overlapping voxels. As a result, significantly more samples from planar surfaces were found if overlapping voxels were applied. In addition, it was found that the threshold value for detecting planarity needs to be adjusted when the voxel size is changed. The trend was that the threshold needed to be increased when the size of voxels decreased.

1. Introduction

UAV (Uncrewed Aerial Vehicle) systems with mounted cameras have become popular due to their ability to provide an aerial perspective at competitive prices. Photogrammetric methods applied to UAV imagery provide dense 3D point clouds with RGB colors. Such data have become popular in the AEC (architecture, engineering and construction) industry (He et al. 2024), archaeology (Günen et al., 2024), cultural heritage (Yiğit and Uysal, 2024), urban planning (Sang, 2024), and monitoring of landscape and environmental states, structures, processes, changes, and hazards (van der Sluijs et al., 2024), just to name a few.

From 3D point clouds, it is possible to search for geometrical primitives such as points, lines, planes, polygons, circles, spheres, and cubes to further analyze and understand the scene. Planes can be utilized for various tasks. In built areas, dominant planes can reveal structures on buildings that can be utilized for reconstruction or augmented reality. Planar areas can also give valuable information for general scene understanding or, e.g., for robot navigation (Liu et al., 2019). On the other hand, corresponding planar areas can also be utilized to register data from different sources (Jonassen et al., 2024) or in SLAM (simultaneous localization and mapping) applications (Behley and Stachniss, 2018). There have even been attempts to extract 3D planes from a single image with the aid of neural networks (e.g., Liu et al., 2019), but using 3D point clouds is a more robust approach.

There are many alternatives to extract a plane from a set of 3D points. One alternative is to fit a mathematical model using a robust method such as RANSAC (Fischler and Bolles, 1981), Hough transformation (Duda and Hart, 1971), least median of squares (Rousseeuw, 1984), or their variants. Alternatively, planar areas can be extracted using a region growing algorithm that starts from a seed point and examines the neighborhood (Rabbani et al., 2006). The neighboring points are needed for defining a normal vector for an examination point. The neighboring points can be selected using either fixed distance neighbors or k-nearest neighbors. Another approach is to use principal component analysis (PCA) (Jolliffe, 2002) of a local

covariance matrix to reveal the direction of the normal vector, as the eigenvector corresponding to the smallest eigenvalue approximates the surface normal (Pauly et al. 2002). In addition, eigenvalues from PCA can be utilized to distinguish planar areas from non-planar areas. In addition, deep learning methods can detect planar surface areas from 3D point clouds. An example of such segmentation is PointNet (Qi et al., 2017).

A voxel, i.e., a volume pixel, represents only one point in a 3D space. However, in the original 3D point cloud from which voxels are created, many points may locate within one voxel. This enables computation of statistical information or geometric properties that can be attached to a voxel (Yang et al. 2021). Voxels are utilized to represent volumetric data in various fields, such as 3D mapping, navigation, environmental modeling, medical imaging, meteorology, geology, finite element analysis, 3D printing, and 3D games.

Voxel-based methods have been widely utilized to find planar surfaces because the previously mentioned plane detection methods can be applied to points located within a voxel. Liu and Zhang (2023) searched planar surface points to register two LiDAR point clouds with a point-to-plane method using voxels. In addition, they enhanced the method with an adaptive voxelization, in which larger voxels were divided into sub-parts to find smaller parts of planes. Afterwards, too small plane parts were merged with adjacent planes. Supervoxels are groupings of voxels, and Tian and Hua (2024) provide an overview of various approaches for creating supervoxels.

Usually, voxels do not overlap. However, Rönholm et al., (2015) utilized overlapping voxel grids to improve voxel-based outlier filtering from 3D data. Inspired by this, overlapping voxels are examined in this article.

The aim of this paper is to examine the behavior of voxel-based detection and sampling of planar surface areas using both non-overlapping and overlapping voxels. Because the focus is on detecting differences when using voxels of different sizes and parameters, only one plane detection method was included, utilizing PCA of 3D covariance matrices. The process takes samples from planar surface areas, as within each voxel location,

the mean point representing one planar surface point and a normal vector are found. Although the method is applicable to any 3D data, this paper focuses on photogrammetric UAV data.

2. Materials and methods

2.1 Data

The complete data set included 449 oblique aerial images (Figure 1) acquired in 2022 with Sony DSC-RX1RM2 35mm full frame Exmor R CMOS camera mounted in GeoDrone 6 quadcopter. The size of images were 7952 x 5304 pixels. The test area located in Helsinki, Finland. The camera was calibrated using self-calibration, and 3D point clouds were created, all within Agisoft Metashape. The point density on the surface with a good texture was approximately 1000 points per m². For this test, only a part of one building was selected (Figure 2). The longest side of the building model is ca. 43 meters. The back side of the building was not included in this test. Otherwise, the photogrammetric 3D point cloud was unprocessed consisting of ca. 2.7 million points.

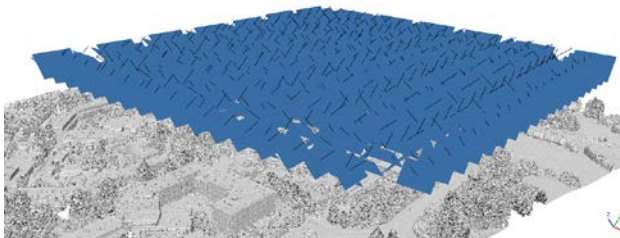


Figure 1. The imaging geometry of oblique UAV images.



Figure 2. The colored dense 3D point cloud of the test building.

2.2 Workflow

All steps were implemented in Matlab. The first step was to compute voxels from a photogrammetric 3D point cloud and to check if a voxel includes a planar surface area. The size of a voxel, which is a cube, was an adjustable parameter. In addition, it was possible to move a voxel with a different shift than the voxel size. In this article, this shift is referred to as 'step size'. This enables the creation of overlapping voxels. The other two parameters of the algorithm were the minimum number of points a voxel must include and the threshold for planarity.

The detection of planar surface areas was based on PCA. For each voxel location, a 3D covariance matrix was computed from the points located inside the voxel. The first step was to compute the mean of all points with Equation 1. The parameter p represents the number of points.

$$\begin{bmatrix} X_{\mu} \\ Y_{\mu} \\ Z_{\mu} \end{bmatrix} = \frac{1}{p} \sum_{i=1}^p \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_i \quad (1)$$

Then, all points were centered according to the mean point.

$$\begin{bmatrix} X_{centered} \\ Y_{centered} \\ Z_{centered} \end{bmatrix}_i = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_i - \begin{bmatrix} X_{\mu} \\ Y_{\mu} \\ Z_{\mu} \end{bmatrix}, i = 1 \dots p \quad (2)$$

If all centered points are stored in a matrix D , where each row includes one centered point, the covariance matrix can be computed with Equation 3.

$$\Sigma = \frac{1}{p-1} D^T D \quad (3)$$

Eigenvalue decomposition was performed on the covariance matrix because the normal vector to the surface corresponds to the eigenvector with the smallest eigenvalue. This decomposition provides three eigenvalues ($\lambda_1, \lambda_2, \lambda_3$). From the eigenvalues, it is possible to establish an equation for surface variation, i.e., a measure of planarity (mp), as presented in Equation 4.

$$mp = \frac{\min(\lambda_1, \lambda_2, \lambda_3)}{\lambda_1 + \lambda_2 + \lambda_3} \quad (4)$$

Setting a threshold for this measure allows identification of planar areas. Only if the measure of planarity indicated that a voxel contained a clear planar surface area, the mean point and the normal vector were stored, providing a sample of the plane. The location of the mean point depends on the distribution of points within a voxel and most likely deviates from the voxel's center point. This method is independent of the orientation of the planar area. However, if the step size is half of the voxel size, it is possible that, in planar surface areas, the same set of points will be selected more than once, leading to duplicate mean points and normal vectors. Therefore, before saving a point, it was ensured that there were no duplicates in the already saved data.

It is possible that some normal vectors need to be flipped to point outwards from the surface. Because the test object in this case was simple, having only three expected growing directions, the flipping was done according to prior knowledge of these directions. A more advanced method would have, for example, involved using an estimated centroid of the object and forcing normal vectors to grow away from it. The complete workflow is illustrated in Figure 3.

2.3 Evaluation methods

For three dominant planar directions, the mean normal vectors were computed. To classify all the found normal vectors into three classes of dominant directions, k-means clustering (David and Vassilvitskii, 2007) was applied. The normal vectors were first normalized by dividing them by their lengths. For each class, the mean normal vector ($\bar{n}$) was computed. For each normal vector (n) in the class, the angle with respect to the mean normal vector was computed (Equation 5). The mean angular deviation was computed by taking the mean value of the absolute values of all angles within a class.

$$\text{angle} = \arccos(n \cdot \bar{n}) \quad (5)$$

In CloudCompare, the longest façade was manually segmented for more detailed examination. The window areas were manually

removed because they contained many uncertain 3D points. Then noise was reduced with the SOR (Statistical Outlier Removal) filter. A plane was fitted to remaining façade points (Figure 4). Figure 5 illustrates how the façade points deviate from the fitted plane, revealing potential planar surface areas as well as some problematic areas. Most of the points are within 3 cm from the plane, but there are some areas where the difference is larger.

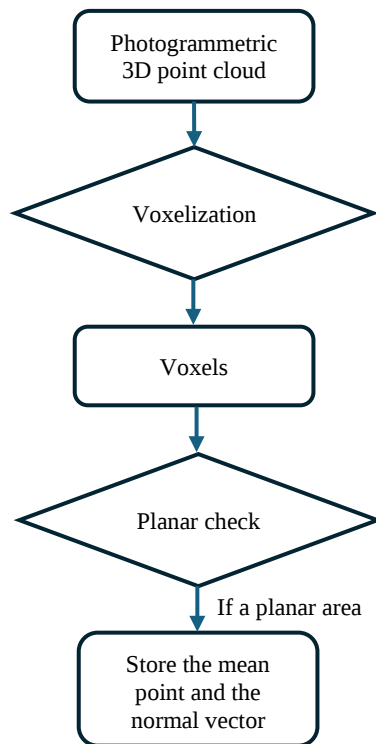


Figure 3. The workflow of the method.

The normal vector of the fitted plane from CloudCompare was compared with the mean normal vector computed from all voxel-based samples of the façade. For this, Equation 5 was applied with the mean normal vector from the voxel-based method and the normal vector of the fitted plane from CloudCompare as the input vectors.

The processing time was recorded for all cases with different parameters. All computations were done on the same computer: Intel i7, 128GB memory, NVIDIA GeForce GTX 1080Ti.

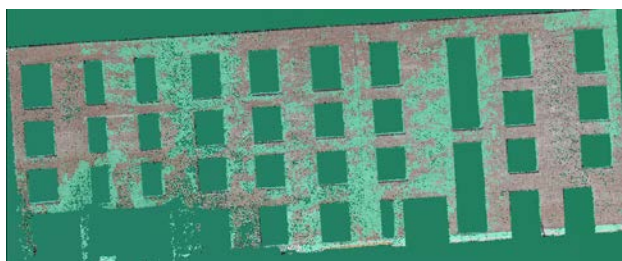


Figure 4. A plane (green) was fitted to potential façade points.

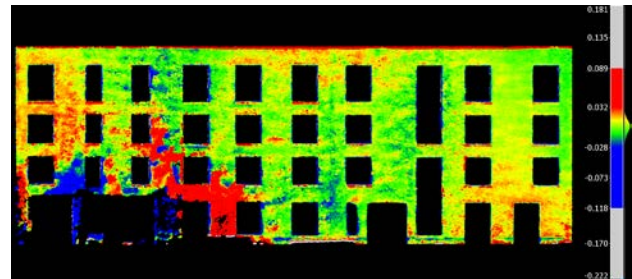


Figure 5. The deviation of the original photogrammetric 3D potential surface points from planarity was observed when compared to the fitted plane.

3. Results

The results in Table 1 used a threshold of 45 for the minimum number of points in a voxel and 0.0001 for the measure of planarity (*mp*). The experiment was repeated with different voxel sizes and step sizes. Table 1 shows the number of found points and the angular deviation to three dominant directions (D1 refers to the direction closest to the X axis, D2 to the direction of the Y axis, and D3 to the direction of the Z axis). In Table 2, the corresponding results using the *mp* threshold of 0.0002 are listed. A larger threshold value means that fewer samples of planar surface areas are accepted.

Voxel size (m)	Step size (m)	Number of points	Mean angular dev., D1 (deg)	Mean angular dev., D2 (deg)	Mean angular dev., D3 (deg)
2.0	2.0	66	0.7	0.5	1.5
2.0	1.0	412	0.8	1.4	1.5
1.5	1.5	163	0.4	0.6	1.6
1.5	0.75	859	1.5	1.2	1.6
1.0	1.0	277	0.9	0.9	1.6
1.0	0.5	1721	1.3	0.7	1.5
0.5	0.5	1010	0.4	0.5	1.2
0.5	0.25	6810	0.3	0.5	0.9

Table 1. The results were obtained using an *mp* threshold of 0.0001 and a requirement that the number of points within a voxel must be 45 or more.

Voxel size (m)	Step size (m)	Number of points	Mean angular dev., D1 (deg)	Mean angular dev., D2 (deg)	Mean angular dev., D3 (deg)
2.0	2.0	100	2.5	0.7	1.5
2.0	1.0	618	1.4	1.7	1.5
1.5	1.5	286	0.9	0.6	1.5
1.5	0.75	1565	1.8	1.4	1.6
1.0	1.0	709	0.8	1.1	1.6
1.0	0.5	3975	1.2	0.9	1.6
0.5	0.5	1923	0.6	0.7	1.6
0.5	0.25	12966	0.6	0.7	1.3

Table 2. The results were obtained using an *mp* threshold of 0.0002 and a requirement that the number of points within a voxel must be 45 or more.

Table 3 illustrates processing times. These values include only the part starting from voxelization and ending when all accepted planar samples and their normal vectors were saved. For

example, plotting the results was not included. Notice that the code was not optimized for speed.

Voxel size (m)	Step size (m)	Processing time (s), $mp < 0.0001$	Processing time (s), $mp < 0.0002$
2.0	2.0	31	31
2.0	1.0	242	237
1.5	1.5	51	50
1.5	0.75	387	377
1.0	1.0	118	112
1.0	0.5	919	893
0.5	0.5	559	519
0.5	0.25	4419	4463

Table 3. Processing times.

In Table 4, the angular mismatch is shown between the mean normal vector of all normal vectors within the selected façade and the normal vector given by CloudCompare for the plane that was fitted to all photogrammetric façade points. In addition to changing the voxel size and step size, two different mp values (0.0001 and 0.0002) were also examined.

Voxel size (m)	Step size (m)	mp threshold	Angular mismatch (deg)
2.0	2.0	0.0001	0.5
2.0	2.0	0.0002	1.3
2.0	1.0	0.0001	0.3
2.0	1.0	0.0002	0.7
1.5	1.5	0.0001	0.1
1.5	1.5	0.0002	0.5
1.5	0.75	0.0001	0.9
1.5	0.75	0.0002	1.0
1.0	1.0	0.0001	0.4
1.0	1.0	0.0002	0.3
1.0	0.5	0.0001	0.7
1.0	0.5	0.0002	0.7
0.5	0.5	0.0001	0.1
0.5	0.5	0.0002	0.2
0.5	0.25	0.0001	0.1
0.5	0.25	0.0002	0.3

Table 4. The effect of the mp threshold on the angular mismatch between the mean normal vector and the normal vector from plane fitting in CloudCompare.

Figures 6, 7, 8, and 9 present visual illustrations of the sampled planar surface areas and their normal vectors. This reveals the distribution and density of the identified planar surface points with their attached normal vectors. The applied voxel sizes were 2.0 m, 1.5 m, 1.0 m, and 0.5 m. In all cases, two threshold values for mp , 0.0001 and 0.0002, were applied.

The results of the comparison between the fitted plane of the selected façade from CloudCompare and the accepted mean points of voxels representing planar surface areas are listed in the Appendix. The comparison illustrates the distances of planar surface samples from the plane.

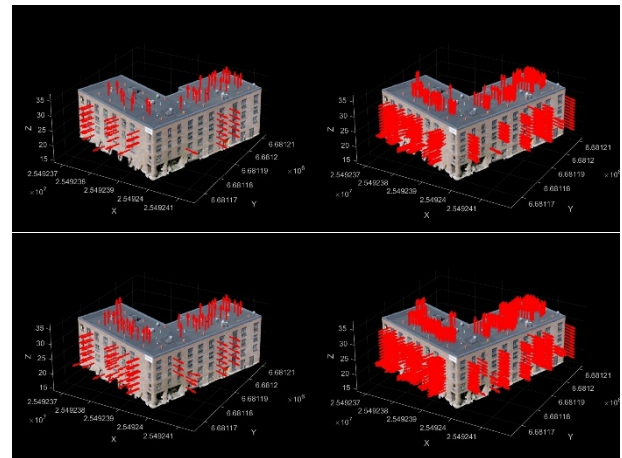


Figure 6. The comparison of found planar samples with normal vectors when the voxel size was 2 meters is shown. In the left column, the step size was 2 meters, and in the right column, it was 1 meter. In the upper row, the mp threshold is 0.0001, and in the lower row, the mp threshold is 0.0002.

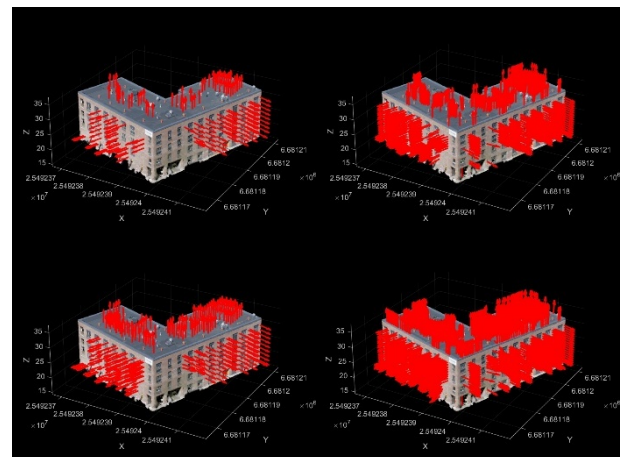


Figure 7. The comparison of found planar samples with normal vectors when the voxel size was 1.5 meters. In the left column, the step size was 1.5 meters, and in the right column it was 0.75 meters. In the upper row, the mp threshold is 0.0001, and in the lower row, the mp threshold is 0.0002.

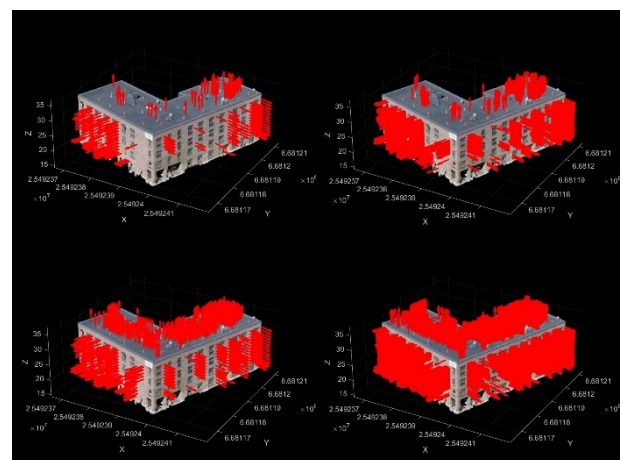


Figure 8. The comparison of found planar samples with normal vectors when the voxel size was 1.0 meters. In the left column, the step size was 1.0 meters, and in the right column it was 0.50 meters. In the upper row, the mp threshold is 0.0001, and in the lower row, the mp threshold is 0.0002.

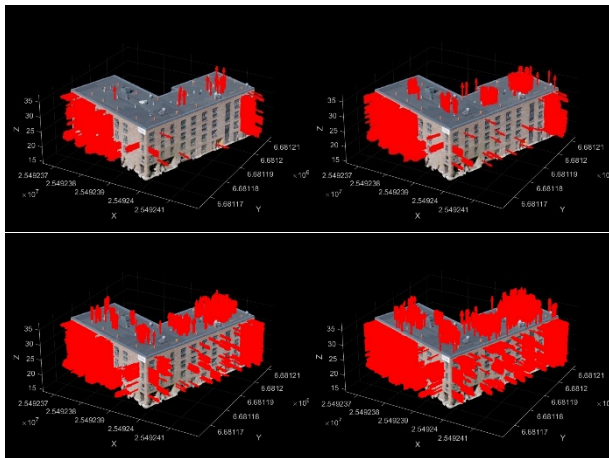


Figure 9. The comparison of found planar samples with normal vectors when the voxel size was 0.5 meters. In the left column, the step size was 0.5 meters, and in the right image it was 0.25 meters. In the upper row, the mp threshold is 0.0001, and in the lower row, the mp threshold is 0.0002.

4. Discussion

The voxel-based method is obviously able to detect planar areas. However, the selection of the voxel size, step size, minimum number of points in a voxel, and the threshold of planarity can significantly change the results. The voxel size sets the limits on how small planar areas can be detected.

When using voxels, there is a chance that potential planar surface areas could be missed due to bad luck. The reason for this is that voxels are usually established in object space and do not account for the structures of objects in the point cloud. Therefore, changing the starting point of the voxel system can result in very different outcomes. This is illustrated in Figure 10, where the black voxel system is unable to detect a potential planar surface area. In contrast, the red voxel system can detect it.

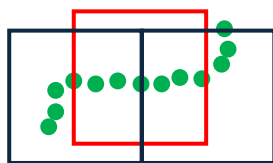


Figure 10. The black voxel system cannot detect the planar area, but the red voxel system can.

Reducing the step size to smaller than the voxel size increases the chances of finding more existing planar surface areas. However, there is a risk that two voxels with different center points may include the same points belonging to a planar surface. Such a case is illustrated in Figure 11. Therefore, the duplicate points were removed in this research. However, if even one 3D point changes, the mean points from adjacent voxels will not be the same. This might lead to a case where there are samples very close to each other. On the other hand, those are still valid samples from planar surface areas. If the number of points in Tables 1 and 2, as well as the illustrations of their distribution in the Appendix, are examined, it is obvious that utilizing a half-voxel step size results in finding significantly more samples from planar areas. A smaller step size is expected to find even more samples from planar surface areas. However, the computing time would increase. If the step size were very short, the method would have no computational advantage over an alternative method that examines local planarity for every point by selecting k -nearest

neighbors (k NN) around an examination point (Barbero-Álvarez et al., 2024).

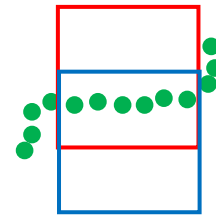


Figure 11. A step size of half a voxel may lead to the selection of identical point sets for two voxels (red and blue) in planar areas.

Processing times were examined and presented in Table 3, and it is obvious that using a smaller voxel size increased processing times. In addition, a smaller step size affected significantly to processing times. In every case, the half-voxel step size led to approximately eight times longer processing time with the current implementation. The code was not optimized for speed. For example, using an octree structure (Elseberg et al., 2013) with 3D point clouds or employing parallel processing would significantly decrease processing times.

Depending on how strict the mp threshold is, this method allows for the detection of planar areas even in the presence of some noise. However, there is no system to detect the direction of noise; the surface point is assumed to be the mean point of all points within the voxel. In cases where noise is directed below or above the surface, the mean point will be biased accordingly.

Noise can also become a limiting issue if a voxel size is small. This can be detected in the case where the voxel size was set to 0.5 meters. Even though significantly more samples of planar surface points were found compared to using larger voxel sizes, the distribution was not as good. One possible reason is that when the voxel size is smaller, the presence of noise becomes more significant. Therefore, the mp threshold should be increased. Indeed, the changing mp threshold from 0.0001 to 0.0002 improved the distribution. However, the significantly longer processing time suggests that using a larger voxel size is better for this point cloud.

In this experiment, PCA was utilized for classifying voxel contents as planar or non-planar surface areas. However, PCA is sensitive to outliers and, therefore, some planar areas might not be detected. Alternatively, robust PCA (Hubert and Rousseeuw, 2005) could be applied to overcome this problem.

The selection of parameters is dependent on the application. If it is crucial to find just a few robust planar surface areas, and it is known that the point cloud includes relatively large planar areas, larger voxel sizes can be applied. An example of such a need is when the aim is to use plane information to solve registration with another data set. If the aim is to detect structural issues, using smaller voxel sizes may be more suitable. In principle, even one found planar surface point with an accurate normal vector can define a large planar area, such as a façade.

It is interesting to further examine the case of a 1.5-meter voxel size and a 0.75-meter step size, which have the worst angular deviation when the mp threshold is 0.0001. The corresponding illustration in the Appendix and comparison with the original data reveal that the distribution looks good overall, but there are a couple of points separate from the others. It appeared that two of these points came from window areas that consist of some unstable points. Therefore, the normal vectors at those locations

are slightly tilted compared to other normal vectors. This suggests that, in this case, the mean angular deviation may reveal if the results include points that do not belong to the same plane.

Another large mean angular deviation can be found in the case where the voxel size was 2.0 meters, the step size was 2.0 meters, and the *mp* threshold was 0.0002. All accepted points are from the façade. However, approximately 22 percent of the points appear to be in the area where the original point cloud does not match the fitted plane exactly (red and blue areas in Figure 5). Those points do not exist in the case where the corresponding voxel size and step size were used but with a lower *mp* threshold (0.0001). This may indicate that the 0.0002 threshold is too high for this voxel size, allowing some uncertain planar areas to be accepted.

Figures 6, 7, 8, and 9, as well as the illustrations in the Appendix, reveal that all variants using the *mp* threshold of 0.0001 found either few or no points from the areas of the photogrammetric 3D point cloud that deviated from the fitted plane. Therefore, this approach, with suitable parameters, could be utilized to detect non-planarity in structures that should be planar. This could be used to detect errors in photogrammetric 3D point clouds. However, in such cases, some prior knowledge about expected planar areas is necessary.

5. Conclusions

In this paper, a voxel-based method to find samples of planar surface areas from UAV-based photogrammetric 3D point clouds was examined using both non-overlapping and overlapping voxels. Overlapping voxels were created by keeping the voxel size constant, but the applied shift of the voxels corresponded to half of the voxel size. As a result, significantly more samples from planar surface areas were found when overlapping voxels were applied.

The effects of different voxel sizes were examined utilizing two different threshold values for planarity. As expected, a smaller voxel size increased computation time but could find more samples from planar areas. In addition, it was found that the threshold value for planarity requires adjustment when the voxel size is changed. The trend was that the threshold needed to be increased when the voxel size decreased. On the other hand, a strict threshold ensures that all found samples represent planar areas. Optimizing the parameters requires further research.

In future work, it would be interesting to test overlapping adaptive voxelization around found samples to efficiently obtain more sample points from clear planar surfaces. In addition, it would be interesting to compare different methods for detecting planarity from the 3D points located within a voxel.

Acknowledgements

The authors would like to thank the Academy of Finland for financial support through the projects “Towards automatic road inspection” (No. 365584) and “Innovative methods for measuring and modelling the dynamic forest road quality” (No. 362926) as well as the European Regional Development Fund project “Circular Green Blocks” (grant number A77489).

References

Barbero-Álvarez, M.A., Jiménez, D., García-Luna, R., Senent, S., Menéndez, J.M., Jiménez, R., 2024. Irregular body surface description using planes defined from noisy 3D point clouds.

Expert Systems with Applications, 237 (part C), Article 121693, 11 pages.

Behley, J., Stachniss, C., 2018. Efficient Surfel-Based SLAM Using 3D Laser Range Data in Urban Environments. In *Proceedings of the 14th Conference on Robotics-Science and Systems (RSS)*, pp. 26-30.

David, A., Vassilvitskii, S., 2007. K-means++: The Advantages of Careful Seeding. In the *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1027-1035.

Duda, R. O., Hart, P. E., 1971. Use of the Hough Transformation to Detect Lines and Curves in Pictures, *Communications of the ACM*, 15(1), pp. 11-15.

Elseberg, J., Borrmann, D., Nüchter, A., 2013. One Billion Points in the Cloud – an Octree for Efficient Processing of 3D Laser Scans. *ISPRS Journal of Photogrammetry and Remote Sensing*, 76, pp. 76-88.

Fischler, M. A., 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), pp. 381-395.

Günen, M.A., Kulakoğlu, F., Besdok, E., 2024. Accuracy Assessment of UAV-Based Documentation of Archaeological Site: Kültepe-Kaneş. *Digital Applications in Archaeology and Cultural Heritage*, 35, Article e00380, 14 pages.

He, T., Chen, K., Jazizadeh, F., Reichard, G., 2024. Unmanned Aerial Vehicle-Based As-Built Surveys of Buildings. *Automation in Construction*, 161, Article 105323, 21 pages.

Hubert, M., Rousseeuw, P. J., 2005. ROBPCA: A New Approach to Robust Principal Component Analysis. *Technometrics*, 47(1), pp. 64-79.

Jolliffe, I.T., 2002. *Principal Component Analysis*. 2nd ed., Springer, 487 pages.

Jonassen, V.O., Kjørsvik, N.S., Blankenberg, L.E., Gjevestad, J. G., 2024. Aerial Hybrid Adjustment of LiDAR Point Clouds, Frame Images, and Linear Pushbroom Images. *Remote Sensing*, 16(17):3179, 22 pages.

Liu, C., Kim, K., Gu, J., Furukawa, Y., Kautz, J., 2019. PlanerCNN: 3D Plane Detection and Reconstruction from a Single Image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4450-4459.

Liu, Z., Zhang, F., 2023. An Efficient Plane Extraction Approach for Bundle Adjustment on LiDAR Point clouds. *Computer Vision and Pattern Recognition*, arXiv:2305.00287, 7 pages.

Pauly, M., Gross, M., Kobbelt, L.P., 2002. Efficient Simplification of Point-Sampled Surfaces. In the *proceedings of IEEE Visualization*, pp. 163-170.

Rabbani, T., van den Heuvel, F., Vosselman, G., 2006. Segmentation of Point Clouds Using Smoothness Constraint. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61(1), pp. 30-114.

Rousseeuw, P. J., 1984. Least Median of Squares Regression. *Journal of the American Statistical Association*, 79(388), pp. 871-880.

Rönnholm, P., Kukko, A., Liang, X., Hyypä, J., 2015. Filtering the Outliers from Backpack Mobile Laser Scanning Data. *The Photogrammetric Journal of Finland*, 24(2), pp. 20-34.

Sang, H., 2024. Application of UAV-Based 3D Modeling and Visualization Technology in Urban Planning. *Advances in Engineering Technology Research*, 12, pp. 912-919.

Tian, P., Hua, X., 2024. Robust Surface Area Measurement of Unorganized Point Clouds Based on Multiscale Supervoxel Segmentation. *IEEE transactions on instrumentation and measurement*, 73, 17 pages.

van der Sluijs, J., Sait, E., Fraser, R.H., Kokelj, S.V., Bakelaar, C.N., 2024. Validation of Beyond Visual-Line-of-Sight Drone Photogrammetry for Terrain and Canopy Height Applications. *Remote Sensing Applications: Society and Environment*, 35, Article 101266, 15 pages.

Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 19 pages.

Yiğit, A.Y., Uysal, M., 2024. Automatic Crack Detection and Structural Inspection of Cultural Heritage Buildings using UAV Photogrammetry and Digital Twin Technology. *Journal of Building Engineering*, 94, Article 109952, 18 pages.

Appendix

The original point cloud of the façade is illustrated in Figure 12.



Figure 12. The photogrammetrically derived 3D point cloud of the test façade.

In the following figures (Figures 13-28), comparisons of the accepted planar surface sample points, obtained using different parameters, are presented. The distances are measured from the façade plane, which was fitted in CloudCompare v2.14.alpha using all potential façade points (Figure 4). The color thresholds for the distances remain consistent across all illustrations.



Figure 13. Voxel size: 2.0 m, step size: 2.0, >45 points, $mp=0.0001$.



Figure 14. Voxel size: 2.0 m, step size: 2.0, >45 points, $mp=0.0002$.



Figure 15. Voxel size: 2.0 m, step size: 1.0, >45 points, $mp=0.0001$.



Figure 16. Voxel size: 2.0 m, step size: 1.0, >45 points, $mp=0.0002$.



Figure 17. Voxel size: 1.5 m, step size: 1.5, >45 points, $mp=0.0001$.

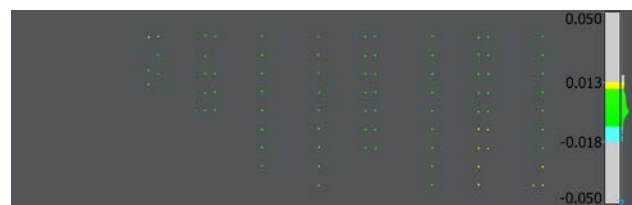


Figure 18. Voxel size: 1.5 m, step size: 1.5, >45 points, $mp=0.0002$.

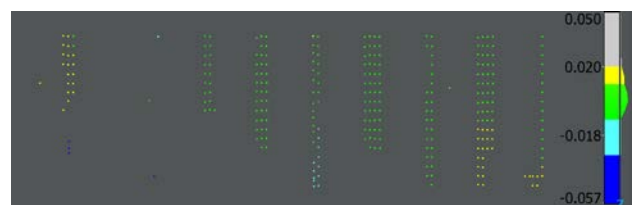


Figure 19. Voxel size: 1.5 m, step size: 0.75, >45 points, $mp=0.0001$.

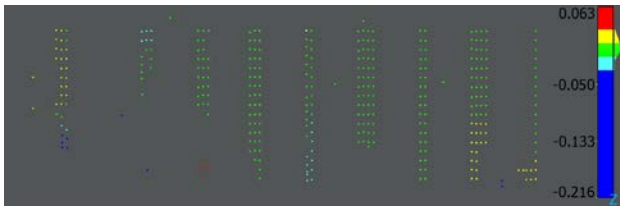


Figure 20. Voxel size: 1.5 m, step size: 0.75, >45 points, $mp=0.0002$.



Figure 21. Voxel size: 1.0 m, step size: 1.0, >45 points, $mp=0.0001$.

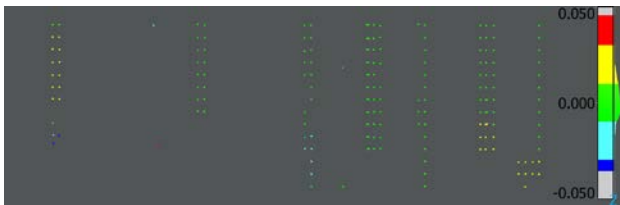


Figure 22. Voxel size: 1.0 m, step size: 1.0, >45 points, $mp=0.0002$.

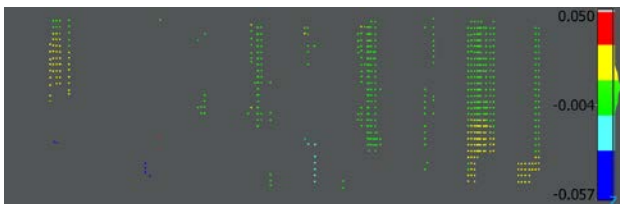


Figure 23. Voxel size: 1.0 m, step size: 0.5, >45 points, $mp=0.0001$.

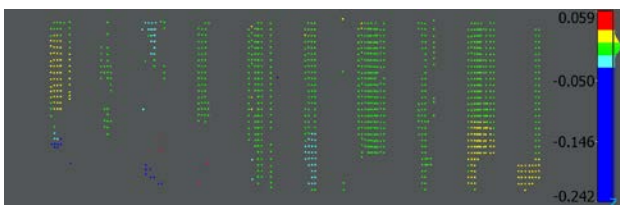


Figure 24. Voxel size: 1.0 m, step size: 0.5, >45 points, $mp=0.0002$.



Figure 25. Voxel size: 0.5 m, step size: 0.5, >45 points, $mp=0.0001$.



Figure 26. Voxel size: 0.5 m, step size: 0.5, >45 points, $mp=0.0002$.



Figure 27. Voxel size: 0.5 m, step size: 0.25, >45 points, $mp=0.0001$.

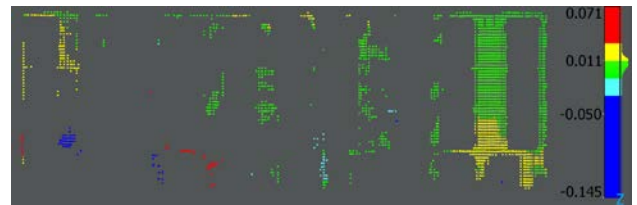


Figure 28. Voxel size: 0.5 m, step size: 0.25, >45 points, $mp=0.0002$.