OBJECT DETECTION AND LOCALISATION FOR BIM ENRICHMENT

Sam De Geyter[a,b] *, Maarten Bassier[a], Heinder De Winter[a,c], Maarten Vergauwen[a]

[a] Dept. of Civil Engineering – Geomatics, KU Leuven – Faculty of Engineering Technology, Ghent, Belgium
[b] MEET HET BV, Mariakerke, Belgium
[c] DIRK BAUWENS NV, Evergem, Belgium
(sam.degeyter, maarten.bassier, heinder.dewinter, maarten.vergauwen)@kuleuven.be

Commission IV, WG IV/4

KEY WORDS: Remote sensing, BIM, Image processing, Object detection, Indoor Mobile Mapping

ABSTRACT:

The use of Building Information Models (BIM) during the entire life cycle of a building or facility requires an up-to-date and detailed digital representation. Nowadays the BIM focus mostly lies on the design and construction phase of the building and is rarely used for the rest of the life cycle. For further use the BIM must be updated after construction and enriched with appliance objects such as safety equipment, heating elements to enhance its usability. In this work a new approach is presented to detect and locate appliance objects in a three dimensional environment using object detection in images created by indoor mobile mapping devices. By detecting and locating these appliance objects in the three dimensional world they can be included in the BIM model. This approach enables the detection and localization needed for the placement of placeholders or detailed geometric models of those appliances in the BIM model or other digital representations. Resulting in an increased level of detail and usability of the digital representation of the building or facility during its further life cycle. Overall, this work demonstrates and examines the accuracy of the detection and localization of appliance objects.

1. INTRODUCTION

The concept of Building Information Models or BIMs focuses on a collaboration between all the buildings' or facilities' stakeholders during its entire life cycle. At the moment most BIM projects are not leveraging the full potential of BIM by only using the BIM concept during the design and build stage of a buildings' life cycle. To increase the usability for later stakeholders it is important that the model gets updated regularly during construction or other life stages, maintaining an up-to-date three dimensional representation of the site. Modern state-of-the-art Indoor Mobile Mapping devices (IMMs) enable this periodical capturing of the onsite conditions in high detail. These devices can capture large sites almost 80% faster than traditional surveying techniques using terrestrial laser scanners while maintaining the accuracy's and even improving the coverage (De Geyter et al., 2022b). Using these techniques also limits the impact of on-site measurements and project planning to a minimum. The output of these measurements can be used for all kind of analyses. For example, a validation of the constructed elements where the as-build conditions are compared to the as-design condition reporting and locating building errors as shown in (Bassier et al., 2023). However, the detailed point clouds and imagery taken by these devices can also be used to update the as-design model to the buildings as-build conditions resulting in an accurate and up-to-date digital as-built representation of the building.

To be of further use during the building's life cycle and unlock BIM's full potential in terms of facility management or safety planning the BIM model should be up-to-date and as detailed as possible at every given moment. These details can be captured using modern remote sensing techniques and should be incorporated in the BIM. This can be done by manually altering already existing elements from the as-design BIM to their as-build state using the remote sensing data or by manually adding new elements to the BIM. Currently manually enriching the BIM with all kinds of appliance objects, adding extra functionalities to the model is a time consuming and labour intensive task. This makes an enriched BIM a very cost inefficient deliverable for the current BIM market. Automating this process by leveraging new techniques in the field of remote sensing and machine learning would make this a highly desired deliverable.

A key challenge to enable an automated BIM creation or enrichment is to interpret the data captured using remote sensing techniques with as little human intervention as possible. To this end, researchers look in the direction of Machine Learning to interpret these huge amounts of data and process them into BIM. Three dimensional semantic segmentation on point cloud data is focusing mainly on large object classes in mostly outdoor environments. Some of these models trained on indoor environments succeed in segmenting classes as walls, floors, chairs, tables, bookcases etc, but these three dimensional semantic segmentation algorithms mostly struggle with the scarce amount of training data available for indoor environments (De Geyter et al., 2022a), occlusions and the limited amount of points on small objects. Additionally creating three dimensional training data is more time consuming than labeling two dimensional images. Two dimensional object detection, on the other hand, is rapidly advancing. The enormous amount of image data, openly available online, combined with training techniques such as data augmentation and transfer learning enable high object detection grades on these data types.

Combining the outputs of state-of-the-art mobile mapping devices and recent advances in machine learning opens the door to an automated approach for a generic appliance object detection and localisation needed for BIM enrichment. Modern IMMs supply an accurate point cloud and a set of detailed panorama images with their corresponding positions and orientations. By positioning each image with a high accuracy using the IMMs SLAM algorithm in the 3D environment of the point cloud the results of two dimensional object detection can be translated to the three

*Corresponding author.

dimensional world.

In this work a novel approach for generic appliance object detection and localisation for BIM enrichment will be presented. In the next section, related work regarding the fields of object detection in both three dimensional as two dimensional data will be provided. Also, recent advances in generic object detection using human input and no custom training will be introduced, followed by the proposed methodology where some limitations regarding the suitability of objects for detection are given. Then, the used detection algorithm and the reprojection methods are proposed. Afterwards the data used for the experiments and the experiments themselves, are discussed, followed by the results of these experiments. To finalise, a conclusion and suggestions for future work are formulated.

2. RELATED WORK

Currently, the use of BIM is mostly focused on the design and construction phase of a building's life cycle. Nevertheless, the use of BIM during the operations and maintenance phase of a buildings' or facilities' life cycle is widely mentioned in literature. The data requirements for BIM models to be useful in facility management on the other hand are not clearly defined in literature.

To unlock the full potential of BIM during the operation stage of a buildings' life-cycle, an enrichment of the BIM model with appliance objects is needed. These appliance objects can vary from fire safety equipment to heating elements and electronic devices. To place these objects in the model, an accurate location of the appliance objects is needed. Additionally, the placement of an object placeholder in the correct location in the BIM model is possible. This placeholder can be an abstraction of the actual object with predefined dimensions, a parametric block adapted to the dimensions of the observed object or a detailed mesh geometry of the object based on the captured point cloud. To enable a cost efficient way of detecting, locating and placing these objects an automated detection an and localisation approach is needed.

The detection of appliance objects from remote sensing data can be done in various ways. Both the 3D point cloud data and the two dimensional image data can be used as input for machine learning algorithms. The use of machine learning for the interpretation of remote sensing data is currently widely researched. Recent machine learning networks are capable of processing the 3D point cloud data directly. The key challenges in processing this 3D data lie in the unstructured nature of this data. These challenges are identified by (Bello et al., 2020) and include the irregularity, unstructuredness and unorderdness of the data. One approach for tackling these problems is to force the point cloud data in a structured representation and exploit the resemblance with two dimensional data. Examples of this approach are voxel- or Multi-view representations. Voxel-based approaches organize the point cloud in a voxel representation with a fixed size. This allows the use of 3D convolving kernels to process the data similar to their 2D equivalent (Maturana and Scherer, 2015, Wu et al., 2015). The high computational cost of this process can be reduced by, for example, only processing the occupied cells from the voxel-grid as presented by (Wang and Posner, 2015). The multi-view approach leverages the existing technology in 2D Convolutional Neural Networks (CNNs) by taking 2D images of the 3D representation from different view points and process them with traditional 2D CNNs (Su et al., 2015). With PointNet (Qi et al., 2017a) the first network able to directly process 3D point clouds was introduced. This network is able to do basic machine learning tasks such as classification, part segmentation and semantic segmentation of point clouds. Its successor includes local features in different scales allowing PointNet++ (Qi et al., 2017b) to take into account the environment of the points and so recognize more fine-grained structures and increase the usability on complex scenes. For object detection on this point cloud data, the data is first passed through a point-based machine learning network to gradually down sample the point cloud data and learn features on different scales (Mao et al., 2023). This approach mainly depends on the number of context points and the used radius for context feature extraction both will have a huge impact on memory consumption.

Object detection using machine learning in images is much more developed. Here the objective of recent studies is to go from detecting a certain type of objects to detecting a very specific object rephrasing the task to a matching problem, and approaching the object detection capabilities of humans (Liu et al., 2020). The field of object detection in images originates from the image classification. Besides the classification of objects, object detection also provides a location for the detected object (Li et al., 2022). The use of machine Learning approaches in the field of object recognition was introduced by AlexNet (Krizhevsky et al., 2012). This work has laid the foundations for object detection in images using deep learning techniques (Li et al., 2022). Two major categories within object detection can be distinguished: one-stage and two-stage object detectors. The main difference between these two approaches is the region proposal step. In two-stage detectors such as R-CNN (Girshick et al., 2014), Fast R-CNN (Girshick, 2015), Faster R-CNN (Ren et al., 2015) an additional screening step is included, filtering the proposals before delivering the final results. Due to omitting this step in one-stage detector networks, these networks have faster processing times. (Liu et al., 2020) identifies the main challenges for deep learning based object detection. First, the large number of classes, different object types and all possible intra-class variations pose a huge challenge for further development on new data. Second, the efficiency of the models with growing computational complexity especially when multiple items in differences scales must be recognised and located, often in one image. Third, scalability is an issue, where networks must be able to detect objects they weren't trained on and preform in situations they haven't seen yet. The author concludes that the growing amount of objects to detect and enormous amount of available data will make annotated training almost impossible, forcing researchers to focus on weakly supervised learning methods.

Both object detection approaches discussed above, whether in 3D or 2D have the same major limitation. All models presented earlier, rely on huge amounts of training data. This training data is mostly handcrafted or relies on a type of manual intervention. This makes the creation of these training data sets a time-consuming and costly process (Liu et al., 2020, De Geyter et al., 2022a).

When targeting a broad variety of objects that can be included in BIM models, (for instance safety signs, heating installations, light fixtures), a more generic object detection approach is needed. Each type of objects also has a huge variety in appearance of different instances, as shown in Figure 1 within the same type, underlining the need for a more generic approach. This almost unlimited set of objects makes the training of a robust object detection network near to impossible. A possible solution can be found in the field of open-world object detection as presented in (Joseph et al., 2021). This method relinquishes the assumption that all classes must be seen during training. To this end, the method must be able to recognize an unknown object as unknown

Figure 1: Tiles from the testdataset containing a emergency exit sign showing the variation between instances of the same type and capturing conditions.

which requires a strong generalisation of the model. The key to this problem is the integration of language into the model. One of the most performing models using this method is GLIP (Li et al., 2021), introducing the contrastive training from (Kamath et al., 2021) on the predicted regions and language phrases. The object detection is reduced to a phrase grounding task. Because of the high resemblance between open-set and closed-set detectors (Liu et al., 2023) expects strong closed-set detectors to result in high preforming open-set detectors. To this end, they use the DETR based model DINO (Zhang et al., 2022) and transform it into an open-set object detector named Grounding DINO (Liu et al., 2023). Due the transformer based architecture of the DINO network, the similarity with language based models is high, enabling an easier integration of both. Also, the overall grounding model is simplified by the end-to-end optimization without hand-crafted modules. The last advantage of the grounding DINO network against GLIP is, because its transformer based architecture, the model is better in leveraging large scale datasets. The grounding DINO object detector takes human language input which gives it a large potential for appliance object detection. By using human input the algorithm can be easily generalised over different objects covering the large variety of appliance objects.

3. METHODOLOGY

The proposed workflow for detecting and locating appliance objects in IMM data is shown in Figure 2. To avoid problems with memory during the object detection, the panoramas with a size of 4096x8192 pixels are cropped into tiles of 512x512 pixels before feeding them to the object detection algorithm. The depth maps corresponding to each sensor position are rescaled to have the same dimensions as their corresponding panorama image, allowing easy mapping between panorama and depth map. The cropping of the panoramic images is done by using the image utility functions embedded in the GEOMAPI API[1]. All data and information is stored using linked data technologies and contains the link between the considered tile and its original panorama image and depth map. Also, the location and orientation of the camera provided by the IMMs SLAM algorithm is stored for the later reprojection.

3.1 Object Detection

The presented approach leverages the use of object detection in images and the data captured by modern IMMs to detect and localise objects in a 3D environment. This is achieved by conducting a custom object detection on the tiles created from the panorama images. The object detection algorithm used in this work is the Grounding DINO object-detector as introduced in section 2. As mentioned before, the panoramas are subdivided in tiles of 512x512 pixels before they are inputted in the network, to reduce memory requirements and to preserve the high detail of the panorama images. This step is especially important when searching for small objects. Each detection made by the algorithm returns a probability of being of the targeted class. All detections with a probability above a certain threshold are stored in a collection D of detected objects, each represented by bounding boxes b. Each bounding box b is defined by four corner points k representing the predicted location of the object in the image and a label l containing the predicted class of the object. To reduce unnecessary computations an optimal probability threshold must be determined, detections with a lower probability then this threshold will be ignored. An experiment to determine the optimal probability threshold is presented in section 5.

3.2 Object localisation

Following the detection of the object in the tile, the detected bounding box on the tile is converted to its corresponding 2D coordinates in the panorama image coordinate system. Afterwards, the 2D image coordinates are converted to 3D coordinates. Since the image is a panorama image stitched from different fish eye cameras, the default pinhole camera model is not applicable. To this end, it is assumed that the image can be projected on a sphere and the 2D image coordinates can be represented by spherical coordinates. For simplicity the center of the bounding box c will be computed, this procedure can be repeated for each pixel included in the bounding box if necessary. As shown in Figure 3 the center of the bounding box c has coordinates v_c and u_c in the 2D image coordinate system. This coordinate system is defined by the u-axis and the v-axis with origin in the left top corner of the image. Assuming the spherical model the distances v_c and u_c can be converted to an angle θ_c in the xz-plane and an angle ϕ_c in the xy-plane, using equations 1 and 2.

$$\theta_c = \frac{\pi(h - 2v_c)}{2h} \tag{1}$$

$$\phi_c = \frac{-2\pi(w - u_c)}{w} \tag{2}$$

In the 3D coordinate system, where the x-axis equals the heading of the IMM and the center of the image represents the origin. Using θ_c and ϕ_c, the 3D coordinates can be computed using equation 3. The depth d_c, which is needed to scale the whole, is equal to the pixel value of the corresponding pixel in the depth map with the same position in the image.

$$c = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} d_c * sin(\theta_c) * cos(\phi_c) \\ d_c * sin(\theta_c) * sin(\phi_c) \\ d_c * cos(\theta_c) \end{bmatrix} \tag{3}$$

Before the detected objects represented by the center c can be inserted in a digital representation of the scene, the coordinates must be converted from the panorama coordinate system to its origin in the panoramic center to the coordinate system of the dataset. This conversion is done using the position and orientation of the capturing position of the panorama provided by the IMMs SLAM algorithm. This capturing position is assumed to be the same as the panoramic center. The position and orientation are provided by a transformation matrix T_{SLAM} which is

[1]GEOMAPI API Python Library: https://geomatics.pages.gitlab.kuleuven.be/research-projects/geomapi/

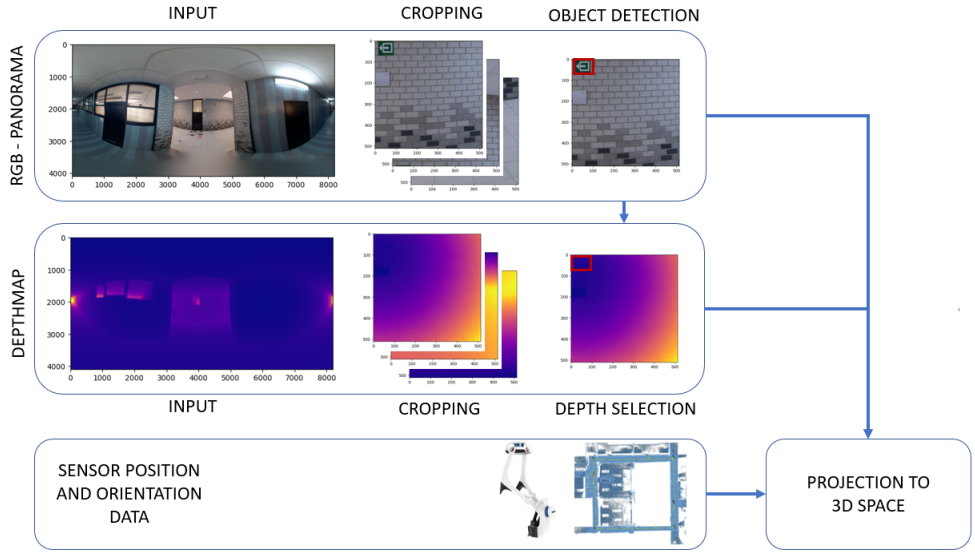

Figure 2: Overview of the proposed workflow showing the three types of data input, RGB-panoramas, depthmaps and position and orientation of the capturing position provided by the IMMs SLAM algorithm. After object detection on the cropped panorama tile, all information is merged to reproject the pixels within the detected bounding box in 3D space.

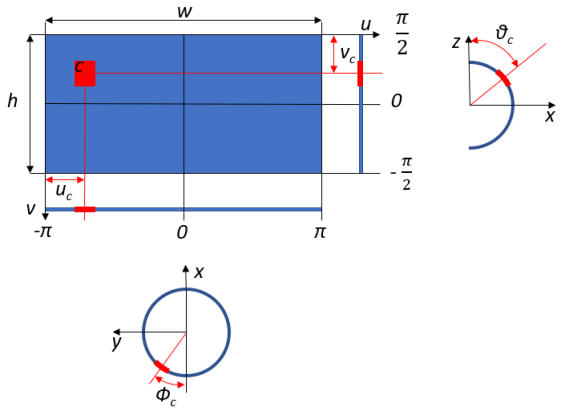

Figure 3: Conversion from 2D image coordinates to 3D coordinates.

optimised during the SLAM post processing steps. Applying this transformation on the computed coordinates of c, shown in equation 4, yields the coordinates in the same local coordinate system as the captured point cloud.

$$c_{local} = T_{SLAM} c \tag{4}$$

When only using this captured point cloud as visualisation, these coordinates of c_{local} are sufficient to represent the point of interest. In most cases these points should be combined over multiple datasets or be presented in other software's or representations such as the BIM. To that purpose, an additional transformation is needed to convert those local coordinates to a reference coordinate system, such as Belgian Lambert 72. The transformation T_{ref} can be determined using multiple methods. One method is by performing a manual registration between the BIM as reference and the captured point cloud. Another method is using known reference points and including these points in the trajectory of the IMM during capturing. By providing the known coordinates of these points to the SLAM post processing, the transformation T_{ref} is computed during the post processing. Using equation 5 the coordinates of C are known in the reference coordinate system.

$$c_{ref} = T_{ref} T_{SLAM} c \tag{5}$$

By using the mobile capturing setup, each object is captured from different viewpoints. This enables the possibility to compare the different detections of the same object. Considering the collection of different images I each image has a set of detected objects D_i (Eq. 6). Selecting all objects where the label corresponds to the same class results in a collection $D_{i,c}$ (Eq. 7. Each of the pixels in these bounding boxes can be represented their center c as computed in equation 5. This results in a collection C_c off all object center points c_i of a certain class with label l_c (Eq. 8).

$$D_i = \left\{ b_d, l_d \middle| \forall i \in I \right\} \tag{6}$$

$$D_{i,c} = \left\{ b_d \middle| \forall i \in I, b_d \in D_i : l_d = l_c \right\} \tag{7}$$

$$C_c = \left\{ c_i \middle| \forall i \in I, b_d \in D_{i,c} \right\} \tag{8}$$

All object centers of the same class which are closer together than a threshold t_d are considered as different occurrences of the same object from different images. This results in a collection of center points C_o representing the same object (Eq. 9).

$$C_o = \left\{ c_i \middle| \forall c_i \in C_c, c_j \in C_c : c_i \neq c_j \cap \|c_i - c_j\| \leq t_d \right\} \tag{9}$$

Finally the position of the objects' point of interest can be computed by computing the mean of all centers in the collection C_o.

3.3 Extracting the 3D object

Performing this procedure on each corner point k for all predicted objects b in the collection D provides a per object plane in 3D space. To extract the points of the point cloud P that are likely to be part of the object, a 3D bounding box is needed. This 3D bounding box is created by offsetting the already found plane towards the capturing position with a distance d_{offset}. This distance d_{offset} defers for each object, at the moment this parameter is set to 1m. This results in collection K per detected object of 3D cubes K_i (Eq. 10) . By extracting all points p_j of point cloud P that are contained by K_i a point cloud P_i can be extracted per

object (Eq. 11).

$$K = \left\{ K_i \middle| \forall b \in B \right\} \tag{10}$$

$$P_i = \left\{ p_j \middle| \forall p_j \in P \cap K_i, K_i \in K \right\} \tag{11}$$

This point cloud only containing the detected object and its direct surroundings allow more detailed analyses. This point cloud can be used for executing part segmentation or parametric object fitting to extract more details about the detected object. In addition, after noise and outlier removal a meshing algorithm can be used to generate a detailed and truthful representation of the object.

3.4 Enriching the BIM

When both the object type and the location of the object are known, the object can be placed in the BIM model. This can be done by inserting a placeholder or a point of interest into the model on the exact location the proposed workflow has located the object. To increase its usability it is possible to link the object to another host object such as a wall or ceiling. To this end a geometric analysis of the distance between the object and these possible host objects can be conducted but is part of future work.

4. DATASET

The site used for testing the presented algorithm is a building located on the KU Leuven university campus of Ghent-Rabot. The remote sensing data used as input for the presented pipeline is captured using the NavVis VLX indoor mobile mapper. This technique allows to capture a high quality point cloud, high resolution panoramas and corresponding depth maps of the site with their corresponding positions in the site's coordinate system. The building has a surface area of approximate 950 squared meters and consists of 4 different floors, each consisting of one room and a common hallway with stairs. On ground level a building physics lab is located. The first and second floor each contain a classroom and the top floor houses a technical room.

The measurement campaign with the IMM covering all floors of the building in one take took about 30 minutes. It resulted in 135 panorama pictures with their position and orientation and corresponding depth maps. During capturing sufficient loop closures were foreseen to maintain good accuracy without control points. In this way, the coverage of the site is optimised and the panoramas provide multiple viewing points for almost each object. After removing points caused by reflections through the windows the point cloud has a size of 2.1 GB and contains approximately 79M points.

4.1 Appliance object selection

Before selecting the appliance objects for detection, some considerations are made. First, the objects need to be useful during the further life cycle of the building and must have a use case in facility management, safety planning, refurbishment or other studies or phases of the buildings life cycle. Second, it is important to be aware that only objects with a direct line of sight to the sensor can be captured and thus detected. Due to the mobile character of the used capturing technique (capturing 360 images every 2m) the coverage of the scene is maximised as much as possible. Nevertheless, some objects are typical hidden from sight. Power outlets, for example, will often be hidden by furniture or electronic devices which makes them invisible for remote sensing techniques. To this end, a visibility study is conducted where the number of occurrences of an object type are compared with the number of occurrences in the panoramic images. Based on this study an object type for further testing is selected.

Table 1: Summary of the number of instances of appliance typical objects present in the testdataset and how many times they occur in the panoramic images taken by the IMM.

Object	# instances	# pano occurrences
Emergence exit signs	16	178
Fire extinguishers	3	66
Fire extinguisher signs	4	69
Fire Alarm buttons	6	85

The test dataset contains different kinds of appliance objects, such as school materials, computers, power outlets, safety signs etc. The visibility study shows that only 70% of all power outlets are visible in the captured data. If these objects need to be modeled it is necessary that the additional 30 % of objects is located and reported manually, reducing the gained time and decreasing the efficiency of the process. In addition, objects in the background, far away of the sensor, will result in a low resolution on the image making their detection impossible.

Because of these limitations the focus in this work is to demonstrate the proposed algorithm on safety equipment. More specifically emergency exit signs are targeted because these objects are intentionally placed in highly visible places making them visible in a maximum number of panorama images. The test set contains 16 of these emergency exit sign instances of 4 different types as shown in Figure 1. These occur 178 times in the captured panoramas. Each instance can be seen in an average of 11 images. Other classes like fire extinguishers, fire extinguisher signs and fire alarm buttons are fare less represented as can be seen in Table 1. Within the set of images large variations in distance to the sensor (size of the object), background, position and light conditions are present, representing a typical mobile mapping dataset as shown in Figure 1.

5. EXPERIMENTS

In a first experiment an optimal probability threshold is determined to input in the grounding DINO algorithm and reduce the number of false positives but still detect all instances. To this end, tests are conducted using different probability thresholds and reporting the number of positive instance detections. For these detections the number of true positive instance detections is compared with the number of false positive instance detections in section 6. To evaluate the detection rate of the targeted objects, the detection results are sorted to determine the precision, recall and F1-score. This analysis is expanded to gain a better insight, by computing per instance metrics including the mean number of tiles each object instance is detected on.

A second experiment tests the accuracy of the proposed reprojection algorithm using ground truth data. After the detection, each pixel of the detected bounding box is projected into the three dimensional space of the original point cloud. This allows a comparison of the projected patch of point cloud to the original point cloud. To this end, a part of the original point cloud containing only the object of interest is separated and considered as ground truth. Both point cloud parts are then compared using cloud-to-cloud distances. This analysis can be done on both the reprojected point cloud per instance or for each detection separately. Both results are shown in section 6. The analysis on the instance point cloud gives an idea of the per instance accuracy achievable using the presented method, whereas the per detection analysis gives a better understanding of the influence of the tile position and angle.

Table 2: Detection rates on tiles using different probability thresholds for Emergency Exit sign detections.

Threshold	Precision	Recall	F1-Score	Accuracy
0.4	11.91%	75.54%	20.57%	92.14%
0.5	26.43%	59.66%	36.63%	97.22%
0.6	50.20%	53.65%	51.877%	98.66%
0.7	72.48%	46.35%	56.54%	99.04%
0.8	95.00%	24.46%	38.91%	98.96%

6. RESULTS

After cropping the 135 panorama images of the dataset in tiles of 512px by 512px, 17.280 tiles are created, each with its corresponding depth map. In 202 tiles an emergency exit sign can be found. After the cropping stage, all tiles are processed by the machine learning algorithm grounding DINO. The search term used for the algorithm is "green exit sign with white arrow" which was experimentally determined beforehand.

First, some tests searching for an optimal probability threshold to assess the results of the grounding DINO algorithm were conducted. In Table 2 the precision, recall, F1-score and accuracy are given for different probability thresholds, considering each tile containing a part of an emergency exit sign as a positive. This table shows the highest accuracy of 99.04% using a probability threshold of 0.7. In Table 3 the same analysis is done for each emergency exit sign instance, presenting the precision, recall, F1-score and the mean number of tiles in which an instance is detected. These results show that using a threshold of 0.7, all instances are detected in on average 6.75 tiles. Examining the False negative results using the 0.7 thresholds shows that the algorithm mostly mislabels other signs or posters present in the scene. These mislabeled objects are only recognized in 2.41 tiles on average and are so clearly distinguishable from the true positive detections. Undetected signs are mostly those pictured in a sharp angle between the camera and the object and are thus located towards the side of the panoramas.

The second experiment shows that the result highly depends on the background of the detected sign within the detected bounding box. As shown in Figure 4 the signs without a clear background result in large point clouds containing points from the background. In most cases these are signs attached to the ceiling or placed on a window. In the case shown in Figure 4 the cloud-to-cloud distance to the ground truth corresponding to 95% inliers is 3.77m which makes it impossible to accurately locate the sign. In Figure 5 an emergency exit sign located on a flat surface (wall) is presented showing the cloud-to-cloud distance corresponding to 95% is around 4cm in this case, and thus yielding more promising results. This difference is mainly explained by the number of background pixels and thus reprojected points within the detected bounding box.

The same conclusions can be drawn when looking at each tile containing a part of the detected instance. An overview of the tiles showing the instance from Figure 5 is given in Figure 6. In this case the cloud-to-cloud distance corresponding to 95% inliers varies between 0.014m and 0.043m with a mean around 0.020m, showing a clear difference between tile configurations. As shown in Figure 6. tiles containing less background clearly have a better result than tiles with more background within the detected bounding box. This is directly influenced by the angle to the cameras' heading and thus the position of the object in the panorama. Objects located in the center of the panorama image have a better detection and reprojection than objects located at the sides, which are more subject to distortion. This distortion clearly impacts the

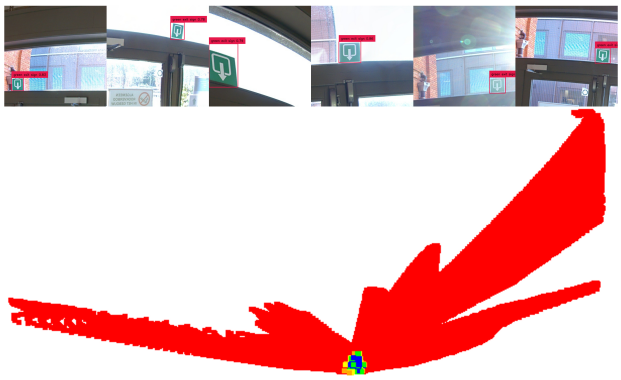

Figure 4: Detected tiles of an emergency exit sign on a window and the reprojection point cloud compared to the ground truth. When no clear background surface is present the reprojected point cloud does not clearly represent the object. The point cloud is colored according to the cloud-to-cloud distance to the ground truth, smaller than 0.005m (blue), smaller than 0.010m (green), smaller than 0.015m (yellow), smaller than 0.020m (Orange), higher than 0.020m (red)

reprojection mainly because of the rectangular bounding boxes which are detected. The influence of the distortion on the outer sides of the panorama apparently remains relatively limited on the detection itself.

For the moment the proposed algorithm is limited to detecting objects with a clear background with an accuracy around 0.04m. Several adaptations of the algorithm can help to overcome this shortcoming and increase the localisation accuracy. For example the outer parts of the panorama image can be ignored, disregarding the zones with large distortions. Another approach can be to increase the probability threshold given to the grounding-DINO algorithm, which will result in less detections. This will remove most detections of signs in the distorted regions of the panorama image. Both methods will directly impact the number of detections, which would more likely lead to a lower recall and F1-score and increase the possibility of missing instances. A more desirable approach can be to use weights, where detections located more to the center of the panorama have more influence to the final result than detections to the sides. The best approach would most likely be to filter out the background points after the detection. This can be done by running a segmentation algorithm on the detected bounding boxes and remove the background before the reprojection. It is important to notice that a part of this inaccuracy will remain, because in the preprocessing of the data the depthmap used for the reprojection is upsampled to have the same size as the panorama image, which can result in some depth inaccuracies on the object edges. Additionally the reprojection algorithm assumes the panorama to be projected on a perfect sphere, which will most likely be not the case.

7. CONCLUSION

The use of BIM during an entire building's life cycle remains a topic for further research. The modern generation of IMMs systems are able to capture a building in high accuracy and detail in a fraction of the time needed by traditional laser scanning setups. This allows for a periodical capturing to update the BIM during the building's usage. The interpretation of the huge amounts of data captured by these devices can be automated using recent machine learning techniques. Employing these techniques for automated data interpretation and BIM creation and enrichment will increase the efficiency of the scan-to-BIM pipeline and impact

Table 3: Detection rates of emergency exit sign instances using different probability thresholds.

Threshold	Precision	Recall	F1-Score	Mean # TP tiles	Mean # FP tiles
0.4	1.80%	100.00%	3.53%	11.00	1.49
0.5	7.96 %	100.00%	14.75%	8.69	2.09
0.6	25.00%	100.00%	40.00%	7.81	2.58
0.7	48.48%	100.00%	65.31%	6.75	2.41
0.8	83.33%	93.75%	88.23%	3.80	1.00

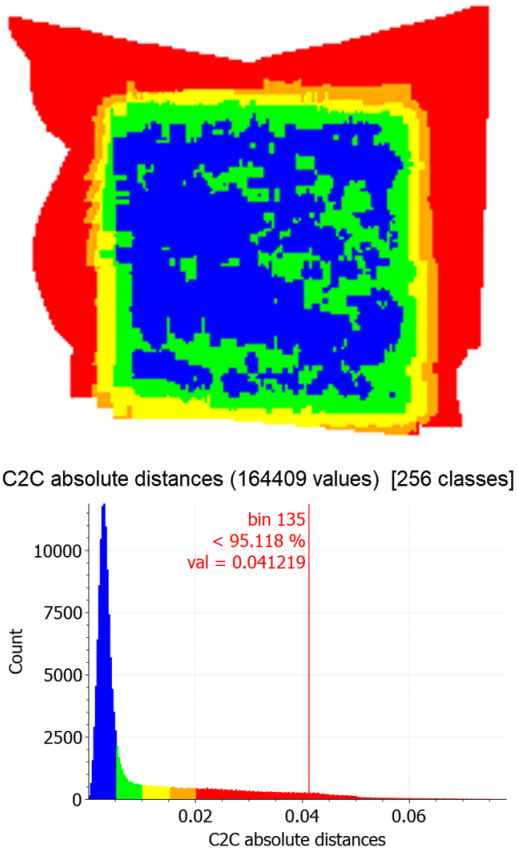

Figure 5: Reprojection point cloud compared to the ground truth of an emergency exit sign with a clear background surface and histogram with the cloud to cloud distances to the ground truth, showing a 95% inlier distance of around 0.04m. The point cloud is colored according to the cloud-to-cloud distance to the ground truth, smaller than 0.005m (blue), smaller than 0.010m (green), smaller than 0.015m (yellow), smaller than 0.020m (Orange), higher than 0.020m (red)

the BIM usage throughout a building's life cycle. Leveraging the rapid advances in the field of 2D object detection on the images taken by modern IMMs and their corresponding location, appliance objects can be detected and located in three dimensional space. This detection and localisation combined with a generic object detector enables the way to enriched BIM models for facility management and safety planning.

The proposed workflow to use state-of-the-art generic object detection algorithms on parts of the panorama images taken by IMMs to localize appliance objects in a 3D environment is promising. The presented work shows that each instance of emergency exit signs in the test building is detected. It is found that the location of the object in the panorama clearly influences the result of the reprojected location. The main problem is the presence of background pixels within the detected bounding box. This problem is most apparent in cases were no clear background is located behind the object. When looking at cases were a clear background behind the object is present the location of the emergency exit sign can be found with a cloud-to-cloud 95% inliers distance of approximately 0.04m. This is sufficient for facility management and safety purposes. Taking into account the errors made by upsampling the depth map during the preprocessing and the assumption of a perfect spherical panorama projection, this error can even be lowered. Segmenting the pixels within the detected bounding box could possibly increase the result by removing these background point before the reprojection.

Future work should examine the possibilities regarding the accuracy of the detection and reprojection. These adaptations, such as removing large parts of the background, would significantly impact the final result. Additionally, the upsampling of the depth maps during the preprocessing and the assumed panorama model should be further examined to gain better insight in the error propagations. Furthermore, tests on different kinds of objects should be conducted. Future works concerning the enrichment of BIM models with appliance objects should examine ways to integrate this gained knowledge into the BIM. Some obstacles here are replacing the points only indicating the location, with more accurate representations. To this end, existing object libraries can be used to determine best fitting library objects to represent the detected object. Also, more parameters can be extracted from the point cloud data. For example when a fire extinguisher is detected in an image, the point cloud of this instance can be separated. The size or volume of the fire extinguisher could then be determined from these points. Other future work includes a detailed study of the required size and number of occurrences of objects, which can directly impact the recording time on site. Additionally, the generality of the object detector should be optimized to reduce the number of false positives which are currently manually removed.

ACKNOWLEDGEMENTS

This project has received funding from the VLAIO BAEKELAND programme (grant agreement HBC.2020.2819) in collaboration with MEET HET BV, the VLAIO COOCK project (grant agreement HBC.2019.2509), the VLAIO BAEKELAND programme (grant agreement HBC.2022.0153) in collaboration with BAUWENS NV, the FWO Postdoc grant (grant agreement 1251522N) and the Geomatics section of the Department of Civil Engineering at the KU Leuven in Belgium.

REFERENCES

Bassier, M., Vermandere, J., Geyter, S. D., Winter, H. D. and Vergauwen, M., 2023. GEOMAPI : Processing remote sensing data with semantic web technologies. Automation in Construction pp. 1–42.

Bello, S. A., Yu, S., Wang, C., Adam, J. M. and Li, J., 2020. Review: Deep learning on 3d point clouds. Remote Sensing 12, pp. 1–34.

De Geyter, S., Bassier, M. and Vergauwen, M., 2022a. Automated Training Data Creation for Semantic Segmentation of 3D

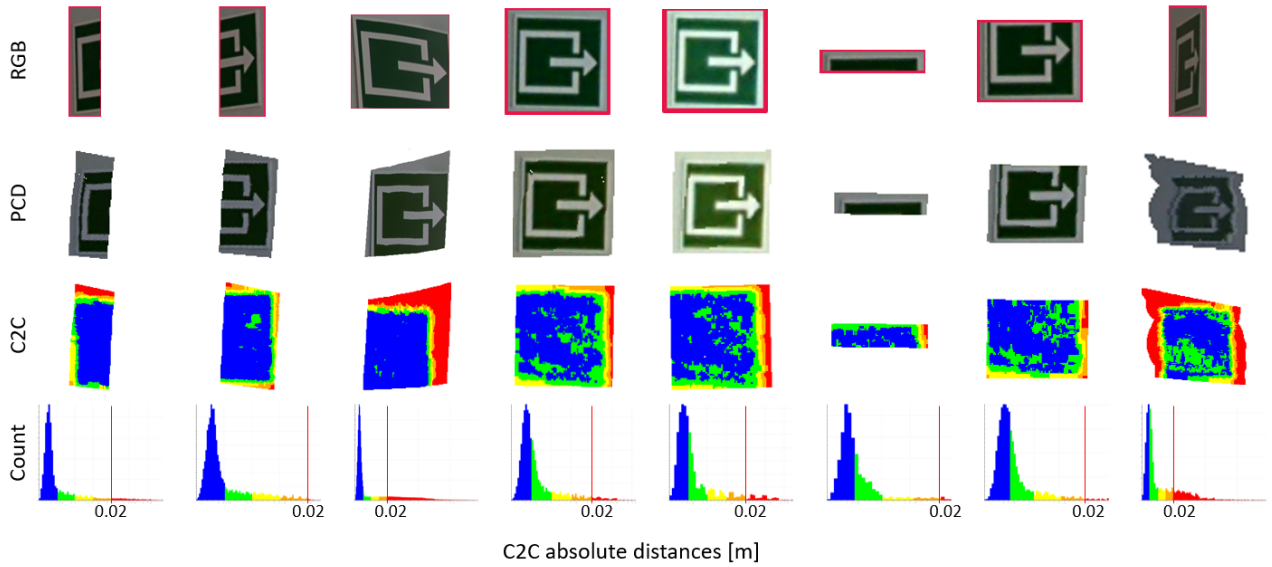

Figure 6: Tiles with a positive detection, their reprojection point cloud compared to the ground truth of an emergency exit sign with a clear background surface and histogram with the cloud-to-cloud distances to the ground truth. The point cloud is colored according to their cloud-to-cloud distance to the ground truth, smaller than 0.005m (blue), smaller than 0.010m (green), smaller than 0.015m (yellow), smaller than 0.020m (Orange), higher than 0.020m (red)

Point Clouds. In: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. Volume XLV, Prague, Czech Republic, pp. 59–67.

De Geyter, S., Vermandere, J., De Winter, H., Bassier, M. and Vergauwen, M., 2022b. Point Cloud Validation: On the Impact of Laser Scanning Technologies on the Semantic Segmentation for BIM Modeling and Evaluation. Remote Sensing.

Girshick, R., 2015. Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision, pp. 1440–1448.

Girshick, R., Donahue, J., Darrell, T. and Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 580–587.

Joseph, K. J., Khan, S., Khan, F. S. and Balasubramanian, V. N., 2021. Towards open world object detection. pp. 5830–5840.

Kamath, A., Singh, M., LeCun, Y., Synnaeve, G., Misra, I. and Carion, N., 2021. Mdetr – modulated detection for end-to-end multi-modal understanding.

Krizhevsky, A., Sutskever, I. and Hinton, G. E., 2012. Imagenet classification with deep convolutional neural networks. pp. 1097–1105.

Li, L. H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.-N., Chang, K.-W. and Gao, J., 2021. Grounded language-image pre-training.

Li, Z., Wang, Y., Zhang, N., Zhang, Y., Zhao, Z., Xu, D., Ben, G. and Gao, Y., 2022. Deep learning-based object detection techniques for remote sensing images: A survey.

Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X. and Pietikäinen, M., 2020. Deep learning for generic object detection: A survey. International Journal of Computer Vision 128, pp. 261–318.

Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J. and Zhang, L., 2023. Grounding dino: Marrying dino with grounded pre-training for open-set object detection.

Mao, J., Shi, S., Wang, X. and Li, H., 2023. 3d object detection for autonomous driving: A comprehensive survey. International Journal of Computer Vision pp. 1–55.

Maturana, D. and Scherer, S., 2015. Voxnet: A 3d convolutional neural network for real-time object recognition. Vol. 2015-December, Institute of Electrical and Electronics Engineers Inc., pp. 922–928.

Qi, C. R., Su, H., Mo, K. and Guibas, L. J., 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017 2017-Janua, pp. 77–85.

Qi, C. R., Yi, L., Su, H. and Guibas, L. J., 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in Neural Information Processing Systems 2017-Decem, pp. 5100–5109.

Ren, S., He, K., Girshick, R. and Sun, J., 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Vol. 28, Curran Associates, Inc.

Su, H., Maji, S., Kalogerakis, E. and Learned-Miller, E., 2015. Multi-view convolutional neural networks for 3d shape recognition. Proceedings of the IEEE International Conference on Computer Vision 2015 Inter, pp. 945–953.

Wang, D. Z. and Posner, I., 2015. Voting for voting in online point cloud object detection. In: Robotics: science and systems, Vol. 1number 3, Rome, Italy, pp. 10–15.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X. and Xiao, J., 2015. 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1912–1920.

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L. M. and Shum, H.-Y., 2022. Dino: Detr with improved denoising anchor boxes for end-to-end object detection.