Automatic MEP Component Detection with Deep Learning

Automatic MEP Component Detection with Deep Learning

John Kufuor[1], Dibya D. Mohanty[1][0000−0001−8976−3896], Enrique Valero[1][0000−0002−0016−4473], and Frédéric Bosché[1][0000−0002−4064−8982]

The University of Edinburgh, School of Engineering
{s1512601, d.mohanty, e.valero, f.bosche}@ed.ac.uk

Abstract. Scan-to-BIM systems convert image and point cloud data into accurate 3D models of buildings. Research on Scan-to-BIM has largely focused on the automated identification of structural components. However, design and maintenance projects require information on a range of other assets including mechanical, electrical, and plumbing (MEP) components. This paper presents a deep learning solution that locates and labels MEP components in 360° images and phone images, specifically sockets, switches and radiators. The classification and location data generated by this solution could add useful context to BIM models. The system developed for this project uses transfer learning to retrain a Faster Region-based Convolutional Neural Network (Faster R-CNN) for the MEP use case. The performance of the neural network across image formats is investigated. A dataset of 249 360° images and 326 phone images was built to train the deep learning model. The Faster R-CNN achieved high precision and comparatively low recall across all image formats.

Keywords: Scan-to-BIM · MEP · Radiators · Sockets · Switches · Convolutional Neural Network · Deep Learning.

1 Introduction

In Scan-to-BIM research, work on automating object detection has mostly focused on large structural components such as floors, ceilings, and walls, or openings such as doors and windows [5,27,32]. However, the effective maintenance of buildings and other structures requires BIM models that contain many other details including mechanical, electrical and plumbing (MEP) components. In fact, MEP assets account for a large share of building maintenance costs [1]. They, thus, constitute important information to collect for maintenance and renovation. Therefore, there is a clear need to develop Scan-to-BIM technology that extends current capability to MEP components.

Detecting MEP components presents a set of unique challenges. They are generally much smaller than structural components which makes it difficult for object detection models to identify them [23]. MEP assets also have a greater range of variation within classes than structural components do; therefore, an

MEP detector must learn more feature patterns. For example, different brands of radiators will have slightly different markings, valve designs and other characteristics.

Recent developments in deep learning have led to impressive results in the detection of many classes of small objects [23, 24]. If successful, the application of deep learning to detect MEP components in photographic and point cloud data will support their integration into Scan-to-BIM frameworks and ultimately deliver more detailed BIM models.

This paper describes work on using deep learning models to detect MEP components in images, specifically sockets, switches and radiators. In addition, the performance of the object detection neural networks across different image formats, specifically 360° images and standard images collected by mobile phones, is interrogated. One of the main challenges of training deep learning models for MEP detection is that building a large dataset is a time-intensive process. If it was possible for models trained on one image format to infer detections on another this would create the opportunity to aggregate larger cross-format datasets. In addition, this cross-format functionality would make the models useful in a wider variety of settings.

2 Related Work

2.1 Mathematical algorithms

The automatic identification and positioning of MEP components in building interior spaces is a field that has received limited research attention. Methods that have been developed usually only detect a single class of object, e.g. pipes.

Regarding electrical components, researchers have experimented with various techniques for detecting sockets. [12] propose a method that finds the holes of sockets in input images using a feature detector algorithm, then applies a geometric equation to group them into coherent sets. A 3D position of the outlet is then determined using a mathematical algorithm called a planar Perspective-n-Point solver. [29] detect a specific variant of orange on white electrical outlets in images using a colour thresholding technique and geometric filtering. Finally, [15] identify the features of a socket in an image using Gaussian filters and contrast limited adaptive histogram equalisation (CLAHE), then apply thresholding to extract the outlet boxes.

[10] detect ceiling lights in laser scan data. The ceiling of the interior space is segmented using a Random Sample Consensus (RANSAC) algorithm. Then the ceiling point cloud is converted into an image by the application of nearest neighbour rasterisation. A Harris corner detector function finds the fluorescent lights in the image and a Hough transform algorithm identifies the circular form of the standard light bulbs.

In the field of plumbing, much of the object detection research has focused on pipes. [9] propose a method for identifying pipes in cluttered 3D point clouds. Assuming that the curvature of a pipe spool would differ from the surrounding clutter, they filter the point cloud for clusters of points that had a cylindrical shape. Then, using the Bag-of-Features method – a computer vision technique where features are aggregated into a histogram – they compare each potential pipe object to what was present in a 3D CAD file of the scene. The clusters with the highest similarity are registered as being accurate representations of as-built pipe spools.

[36] also apply a curvature-based algorithm to point cloud data in order to detect pipelines. Their system uses region growth to segment the point cloud. Then segments are classified as pipelines based on whether 30 randomly selected points fit a curvature requirement. Alternative research detects pipes in laser scans of interior spaces by applying the Hough transform algorithm to slices of the point cloud [2, 6]. [11] proposed a point cloud MEP detection method that begins with a multi-level feature extraction of each 3D point, including variables such as the curvature and roughness, followed by the development of potential pipe segments from promising 'seed' points.

It is also notable that automated pipe detection in point cloud data is a functionality that a number of commercial software packages already offer.

More recently, [1] built a system that detected multiple classes of MEP components including extinguishers, sockets and switches. First, the system extracts orthoimages of walls from point clouds of interiors spaces. Then these orthoimages are separated into their geometric and colour components. Colour-based detection algorithms are applied to the colour images, and geometric detection algorithms are applied to the depth images. Finally, objects are recognised and positioned based on the consensus between the two results.

2.2 Machine learning

Machine learning, where computer models learn how to perform tasks from experience, has also been employed in research on MEP detection [30].

[20] use a random forest classifier – a type of machine learning classifier – together with a sliding window on orthophotos of walls to detect sockets and light switches.

[17] developed a framework for detecting a range of objects in point cloud data, including MEP assets such as valves and spotlights. Primitive shapes, such as pipes and planes, are identified using a support vector machine SVM) which is another type of machine learning classifier. Large primitives, such as walls, are assumed to be background elements and discarded. Then, the remaining points are clustered using their Euclidean distance. Clusters that passed a linearity filter undergoes a detailed matching process comparing them to components in a pre-made 3D object library. If the alignment between a cluster and a target component exceeds a threshold, the cluster is deemed to be a detected instance of the target.

[21] extracted visual features from images using a 'Histogram of Oriented Gradients' and used them to train a SVM to detect radiators. Based on whether the radiator was present or not present their system could evaluate the progress of installation works.

Alternative data sources, such as thermal imaging have also been used. [19] identify light fixtures and display monitors by applying a machine learning clustering model to a thermal point cloud.

2.3 Deep learning

Deep learning models have outperformed conventional computer vision systems when applied to industry-standard object classification and detection benchmarks such as the Pascal VOC challenge and the MS COCO challenge [13]. In these challenges, models are tasked to detect a wide array of common objects such as cars, humans, chairs and clocks in 100,000s of images [25]. Based on the widely documented success of deep learning in computer vision solutions, applying deep learning technology to the MEP use case should produce significantly better results than prior attempts.

There are already many positive case studies of the application of object detection deep learning models to construction and asset management problems. [3] developed a deep learning model that semantically segmented furniture in laser scans of interior spaces. [7] used a neural network to detect structural elements such as beams and columns in the S3DIS point cloud data set. These successes offer further motivation to investigate the application of deep learning to MEP asset detection.

3 Methodology

3.1 Dataset

Building a large, diverse dataset is the most critical step in any successful deep learning project. By providing a large quantity and variety of examples for the model to learn from, a robust dataset ensures the model will perform well when applied to a range of real-world cases [31]. Currently, there are no open-source image databases focused on MEP assets in interior spaces. Therefore, one of the main objectives of this project was to build a small MEP dataset that could be used to train models using transfer learning.

In fact, we built one dataset composed of two sub-datasets. A Ricoh Theta V 360° camera was used to capture the RGB images that make up the 360° sub-dataset. And a selection of mobile phones (including the Pixel 2, Galaxy S8, and iPhone5s) was used to collect samples for the standard image sub-dataset. A variety of lighting conditions, camera angles and levels of occlusion are represented in the data. The images were collected from different residential and educational (University) buildings in UK. The dataset contains radiators of different models, shapes and sizes, Type-G sockets of different types(single and double sockets) and styles, and switches of different styles. Figure 1 shows the variety of the data in each category. This diversity ensures that models trained on the dataset will generalise well to a range of realistic cases.

The images gathered for the dataset were large and this presented a problem (360° images = 5376 × 2688; standard images = 4272 × 2848 - 1600 × 739).

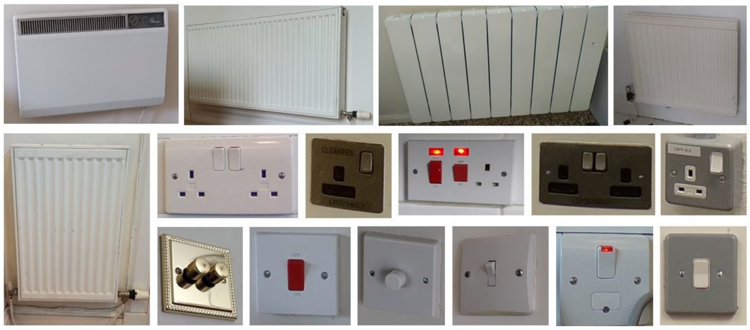


Fig. 1: Sample Images from Dataset

Training the deep learning model on such massive images would be very computationally expensive so they had to be scaled down to a reasonable input size. This process makes sockets, switches and radiators that are already small with respect to the image harder for a model to detect [13]. The amount of detail in the images that the model can use to make decisions is compressed.

To address this issue, the dataset images were segmented into ’tiles’ as shown in Figure 2. Thus, the MEP objects were made larger with respect to their images and the scaled down inputs. Each 360° image was divided into six tiles and each standard image was divided into four tiles. Then the tiles that contained sockets, switches and radiators were selected for use in the corresponding sub-datasets.

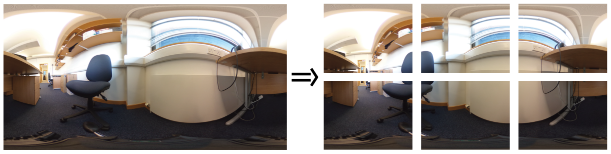


Fig. 2: Tiling

Using a Python program, 80% of images were randomly assigned to the training portion of each sub-dataset, and the remaining 20% were assigned to validation. The sockets, switches and radiators in the images were labelled using the open-source tool LabelImg. Finally, the completed dataset of images and labels was uploaded to Google Drive where it could be accessed from a remote server and used to train models.

3.2 Model Training and Validation

The deep learning model chosen for this research is one with the Faster Region-based Convolutional Neural Network (Faster R-CNN) meta-architecture that uses Neural Architecture Search Net (NASNet) as a feature extractor and was previously trained on the Microsoft Common Objects in Context (COCO) detection dataset. Faster R-CNN has achieved state-of-the-art accuracy on industry standard benchmarks [33].

Model parameters and training process settings were defined in a configuration file before the training began. These included variables such as the input image size the model would accept, the number of training steps that should occur and the location of the dataset files.

During each validation trial, the values of loss, average recall (AR) and average precision (AP) were saved in log files. Once training was complete, the data was used to analyse the model's learning progress, as shown in Section 4.

During the model development process it was observed that growth in performance usually stagnated in the 10,000 to 20,000 training step range. This stagnation signalled that overfitting was occurring and that the model was learning feature patterns that were useful when applied to the training data but did not generalise to new validation data. Thus, the training process for all of the experimental models was stopped at 20,000 steps in order to deliver the highest performance models using the least possible computational resources.

3.3 Dataset Augmentation

Dataset augmentation was used to increase the size of the dataset and thereby improve model performance. Horizontal flip was randomly applied to the images with a 50% probability. Patch Gaussian, was also implemented. This method adds a square of Gaussian noise to a random location on the input image [26]. The square was applied with a 50% probability at random sizes between $300 pixels^2$ and $600 pixels^2$ with a random standard deviation between 0.1 and 1.

4 Experiments and Results

4.1 Dataset

A total of 249 360° images were collected and labelled. When segmented into six tiles as described in Section 3.1, this image base expanded to 1,494 images. 105 of these contained radiators and were selected for the radiator sub-dataset. The 80/20 split resulted in 84 training examples and 21 validation images. 224 images contained sockets and were selected for the socket sub-dataset, with 179 training images and 45 validation images.

326 landscape phone camera images were captured. After being cut into four tiles as described in Section 3.1, the standard image set expanded to 1,304 images. The result was a radiator sub-dataset of 97 images, a socket sub-dataset of 190 images and a switch sub-dataset of 76 images.

4.2 Measuring Performance

Measuring the Average Precision (AP) and Average Recall (AR) is a common strategy for evaluating the performance of object detection neural networks [13, 18, 34]. During each validation step, these metrics are calculated by comparing the detections the model made on the validation images to the labelled reality.

The AP of an object class is the area under the precision-recall curve. AP is defined according to the Intersection over Union (IoU) threshold that was used. For example, mAP@0.5 is the mAP when 0.5 is the IoU threshold for a positive detection.

AR is the maximum recall given a defined number of detections per image, averaged over a range of IoU thresholds, specifically IoU=0.5 to IoU=0.95 with a step of 0.5 [8,16]. For example, AR@100 is the average recall at a maximum of 100 detections per image. Since all the images in our dataset contain more than one and less than ten MEP assets AR@10 is used in the experimental analysis.

K-fold cross-validation was used to verify the accuracy of the results. This strategy is standard practice in leading-edge object detection research [28, 38]. For the following experiments K=3 was used and images were allocated to training and validation at an 80%/20% split.

4.3 360° Image MEP Detector

As shown in Table 1, the maximum AP@0.5 achieved by the Faster R-CNN when trained on the 360° radiator sub-dataset was 0.897. This means that at peak precision an average of 89.7% of the model's detections were accurate.

The AR@10 of the radiator model reached a maximum of 0.728, as seen in Table 2. This metric indicates that, on average, 72.8% of the radiators in the validation images were detected.

The socket model had higher performance in terms of precision with an AP@0.5 of 0.939 but lower recall with an AR@10 of 0.698.

Comparing these values of recall with the relatively high precision, it is evident that although most of the Faster R-CNN's predictions were accurate there were a high number of false negatives where radiators and sockets of interest were missed entirely.

Table 1: Peak AP@0.5 of 360° models

Class	Fold 0	Fold 1	Fold 2	Average
Radiator	0.884	0.960	0.892	0.897
Socket	0.929	0.936	0.951	0.939

Table 2: Peak AR@10 of 360° models

Class	Fold 0	Fold 1	Fold 2	Average
Radiator	0.719	0.735	0.730	0.728
Socket	0.693	0.706	0.694	0.698

4.4 Standard Image MEP Detector

Training the Faster R-CNN on standard images yielded improved accuracy. The peak AP@0.5 of the standard image radiator model was 0.995, as seen in Table 3.

Therefore, the fraction of the detections the model made that were accurate was higher. This could be due to more frequent true-positive detections, fewer false-positive detections, or both. The maximum AR@10 exhibited during validation was 0.886, as shown in Table 4. This is evident that the Faster R-CNN was able to identify and locate a greater proportion of the ground-truth radiators when trained with standard images.

The precision and recall of the socket models was also higher than those trained with 360° images with a maximum AP@0.5 of 0.977 and a peak AR@10 of 0.762.

Table 3: Peak AP@0.5 of standard image models

Class	Fold 0	Fold 1	Fold 2	Average
Radiator	0.985	1.000	1.000	0.995
Socket	0.978	0.963	0.989	0.977
Switch	0.980	1.00	0.965	0.982

Table 4: Peak AR@10 of standard image models

Class	Fold 0	Fold 1	Fold 2	Average
Radiator	0.929	0.855	0.873	0.886
Socket	0.770	0.725	0.791	0.762
Switch	0.844	0.906	0.822	0.857

This improved performance could be the result of a number of factors. The number of training samples collected for the standard image sub-datasets was lower than that gathered for the 360° sub-datasets, therefore the jump in model accuracy must have been a result of the content of the images as opposed to their quantity.

The standard image data may have presented a less diverse range of examples to study and therefore fewer patterns for the model to learn, resulting in improved validation results [35]. The increased accuracy could also be because the randomly selected validation testing examples were too similar to the training examples. This is likely judging from the unrealistically high AP results recorded in some of the experiments. In fact, an AP of 1 which would normally be judged as anomalous was observed when training the radiator models on two of the folds. More K-fold trials could have been used to ensure that training-validation splits that proved too easy for the model did not overly skew the results.

Another explanation is that even though the 360° camera had high pixel resolution, the level of detail in the images was low because the pixels were stretched over a 360° frame [4]. The phone cameras captured a much smaller field of view. Therefore, even if their pixel resolution was lower than the 360° camera a higher level of detail could be achieved for objects of interest. This could explain the superior performance of the models trained on standard images.

The switch models also exhibited strong performance with an AP@0.5 of 0.982 and an AR@10 of 0.857. As was the case for the 360° models, precision was higher than recall in all of the standard model experiments.

4.5 Cross-format Testing

Proving that MEP detection models trained with this methodology can deliver high performance across different image formats would make them useful in a wider range of settings. For the sake of repeatability and fair comparison all the models used for the following cross-format experimentation had the exact same 20,000 steps of training time. The previously explored Peak AP models all had different amounts of training because peak AP was achieved at different stages in the training process for each fold.

It is expected that cross-format functionality should be possible between 360° and standard images because the only major difference between the two formats is the level of distortion created by the 360° camera. The Ricoh Theta V camera used for this research employs two fisheye lenses on the front and back. Fisheye lenses make use of barrel distortion, where object scale decreases with distance from the optical centre, to capture an extended field of view seen in Figure 3 [22].

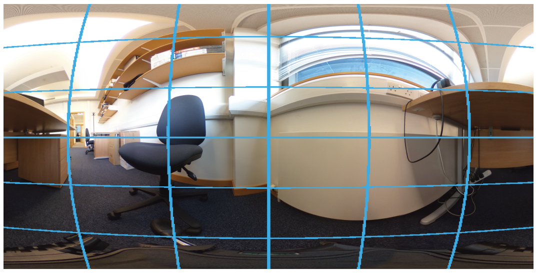


Fig. 3: Barrel Distortion

The distortion that the 360° model encounters during training could potentially help it to fare better when faced with ‘simpler’ image samples. There is evidence that intentionally applying optical distortion to training images for augmentation purposes can improve precision. For example, [39] used different lens distortions, including barrel distortion, to augment image training data for their Deep Image neural network and achieved state of the art classification accuracy.

In support of this theory, testing the 360° radiator model on 360° images and standard images yielded similar results. As shown in Table 5, the AP was only 0.001 lower when the model was applied to standard images.

However, as seen in Table 6, the 360° socket model exhibited significantly lower accuracy when tested with standard images.

Occlusion may have played a significant role in this drop in performance. 31% of the sockets in the standard image dataset are only partially visible in the image tile (note that a fully visible socket with a plug is still considered partially visible), while only 2% of the sockets in the 360° images were partially hidden. As research has demonstrated, objects that are only partially visible are harder for a model to recognise because they offer fewer visual features [14]. The 360° socket model was not prepared to handle a high incidence of occluded objects.

Table 5: AP@0.5 results of testing the 360° radiator model on 360° images and standard images

	360° images	Standard images
Fold 0	0.794	0.874
Fold 1	0.943	0.822
Fold 2	0.847	0.885
Average	0.861	0.860

Table 6: AP@0.5 results of testing the 360° socket model on 360° images and standard images

	360° images	Standard images
Fold 0	0.920	0.661
Fold 1	0.918	0.609
Fold 2	0.941	0.651
Average	0.926	0.640

To verify whether occlusion was actually hindering the performance of the 360° models on the standard images, the 360° models were tested on the uncropped standard images. In their uncropped format all of the sockets in the images were fully visible. As explained in Section 3.1, using uncropped images came with the disadvantage that the MEP assets would be smaller with respect to the images and therefore harder to detect in the feature map. Despite this handicap, the performance of the 360° models on the uncropped standard images was much higher. As shown in Table 7, AP on standard image sockets rose 21% to 0.773.

Interestingly, testing on uncropped standard images also improved the cross-format performance of the radiator 360° model. As shown in in Table 8, AP increased 10% to 0.949. This improvement can similarly be attributed to the fact that most of the radiators are fully visible in the 360° training data and uncropped standard images, but not in the standard image cropped tiles.

Therefore, it can be concluded that the high occurrence of occlusion in the tiled image examples hindered model performance. This indicates that in future research on the use of neural networks for MEP detection, models should be developed with occlusion invariance so that they can better handle such cases. This could be achieved through further diversification of the training data to ensure that the proportion of occluded objects is similar to what is expected in the testing and real-world samples. There are also many examples in published research of deep learning object detection architectures that have been designed to handle occlusion using techniques that could be applied to the MEP context [37].

The standard image models fared much worse when tested on 360° images. As shown in Table 9, AP was 34% lower when the standard image radiator model was applied to 360° images. Similarly, Table 10 shows that AP was 43% lower when the socket model was applied to 360° images. This supports the theory that the barrel distortion of the 360° images better prepares the 360° models to

Table 7: AP@0.5 results of testing the 360° socket model on standard images and uncropped standard images

	Standard Images	Uncropped Standard images
Fold 0	0.661	0.788
Fold 1	0.609	0.731
Fold 2	0.651	0.801
Average	0.640	0.773

Table 8: AP@0.5 results of testing the 360° radiator model on standard images and uncropped standard images

	Standard Images	Uncropped Standard images
Fold 0	0.874	0.950
Fold 1	0.822	0.951
Fold 2	0.885	0.946
Average	0.860	0.949

handle standard images with less optical distortion but hampers the performance of the standard image models when they are tested on 360° images.

Table 9: AP@0.5 results of testing the standard image radiator model on standard images and 360° images

	Standard Images	360 images
Fold 0	0.976	0.615
Fold 1	0.984	0.660
Fold 2	0.996	0.649
Average	0.985	0.641

Based on these results, it can be concluded that in future applications of Faster R-CNN to cross-format object detection, models trained on the format with the most distortion will likely perform the best when applied to other formats.

5 Conclusion

5.1 Summary and Limitations

This paper aimed to investigate the usefulness of deep learning neural networks in detecting sockets, radiators and switches in images. A dataset of 360∘ images and standard phone images was built and used to retrain an existing Faster R-CNN model. Then, an analysis of the deep learning model performance in and across these formats was carried out to explore how best to apply the Faster R-CNN for this use case. The results proved that neural networks can be an effective tool for detecting MEP assets and thereby add value to Scan-to-BIM frameworks.

As discussed in Section 4, the Faster R-CNN exhibited high precision and comparatively low recall when trained on both the 360° images and standard

Table 10: AP@0.5 results of testing the standard image socket model on standard images and 360° images

	Standard Images	360 images
Fold 0	0.973	0.595
Fold 1	0.960	0.539
Fold 2	0.988	0.531
Average	0.974	0.555

images. This indicates that most of the model's predictions were accurate, but there were many false negatives i.e. sockets and radiators that were overlooked. Strategies to overcome this challenge are discussed in the following sub-section.

The primary limitations faced in this research have been high computational demand, and a limited dataset. Training multiple models to cross-validate each experiment was a time intensive process. In further research that builds upon this project, the existing setup could be scaled to make use of a cluster of dedicated GPU or TPU servers. This would facilitate the execution of more detailed experiments exploring a wider range of model configurations.

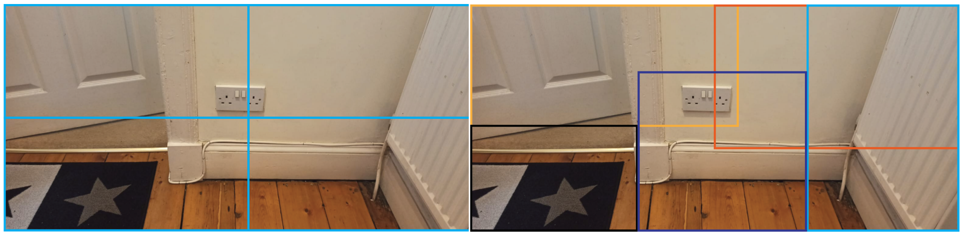
The dataset for this project did not achieve the scale of industry-standard datasets which usually have millions of object instancess [25]. Also, the standard image sub-datasets did not have sufficient differentiation between the training and validation images which resulted in anomalously high validation results. Therefore, this project can only give a limited view of the accuracy that could be achieved in the practical application of Faster R-CNN to detecting MEP components. However, the observations that have been made on the model's precision-recall relationship and cross-format performance in this use case can be applied to future work backed by more resources.

5.2 Future Research

There are many promising methodologies outside the scope of this paper that could be investigated to support the integration of MEP object detection into other processes, such as scan-to-BIM.

The cross-validation experiments detailed in Section 4.5 evidenced that occlusion can significantly hinders model performance. One technique that could be investigated for addressing this issue is the use of overlapping tiles as opposed to the discrete ones used in this project. Using an overlapping tile system as shown in Figure 4, would mean that even if an MEP asset was not fully visible in one tile it would be more likely to appear fully in another. The detections from the overlapping images could then be combined to yield improved overall performance. This is preferable to the discrete tile system where, if one tile contains a portion of an asset, then none of the other tiles can provide a full view of that asset.

Another area of research that should be explored is the combination of detection data from multiple images of an interior space linked through photogrammetric reconstruction. Gathering the classification and location data for all the MEP components in a room using detection models applied to overlapping im-



(a) Discrete tile system

(b) Overlapping tile system

Fig. 4: Comparison of tiling methods

ages and merging this information with 3D photogrammetric data would be useful to improve overall detection performance, as well as to support subsequent processes, for example to automatically insert those assets into BIM models generated through Scan-to-BIM processes.

Acknowledgements

This work was conducted as part of the BIMERR project has received funding from the European Commission's Horizon 2020 research and innovation programme under grant agreement No 820621. The views and opinions expressed in this article are those of the authors and do not necessarily reflect any official policy or position of the European Commission.

References

1. Adán, A., Quintana, B., Prieto, S.A., Bosché, F.: Scan-to-BIM for 'secondary' building components. Advanced Engineering Informatics **37**, 119–138 (aug 2018)
2. Ahmed, M.F., Haas, C.T., Haas, R.: Automatic Detection of Cylindrical Objects in Built Facilities. Journal of Computing in Civil Engineering **28**(3) (may 2014)
3. Babacan, K., Chen, L., Sohn, G.: Semantic Segmentation of Indoor Point Clouds Using Convolutional Neural Network. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences **4**, 101–108 (nov 2017)
4. Barazzetti, L., Previtali, M., Roncoroni, F.: Can we use low-cost 360 degree cameras to create accurate 3d models? International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences **42**(2) (2018)
5. Bassier, M., Vergauwen, M., Van Genechten, B.: Automated classification of heritage buildings for as-built BIM using machine learning techniques. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences. vol. 4, pp. 25–30. Copernicus GmbH (aug 2017)
6. Bosché, F., Ahmed, M., Turkan, Y., Haas, C.T., Haas, R.: The value of integrating Scan-to-BIM and Scan-vs-BIM techniques for construction monitoring using laser scanning and BIM: The case of cylindrical MEP components. Automation in Construction **49**, 201–213 (jan 2015)
7. Chen, J., Kira, Z., Cho, Y.K.: Deep Learning Approach to Point Cloud Scene Understanding for Automated Scan to 3D Reconstruction. Journal of Computing in Civil Engineering **33**(4) (jul 2019)

8. COCO Dataset: Detection Evaluation (2019), http://cocodataset.org/#detection-eval
9. Czerniawski, T., Nahangi, M., Haas, C., Walbridge, S.: Pipe spool recognition in cluttered point clouds using a curvature-based shape descriptor. Automation in Construction **71**(2), 346–358 (nov 2016)
10. Díaz-Vilariño, L., González-Jorge, H., Martínez-Sánchez, J., Lorenzo, H.: Automatic LiDAR-based lighting inventory in buildings. Measurement: Journal of the International Measurement Confederation **73**, 544–550 (jul 2015). https://doi.org/10.1016/j.measurement.2015.06.009
11. Dimitrov, A., Golparvar-Fard, M.: Segmentation of building point cloud models including detailed architectural/structural features and mep systems. Automation in Construction **51**, 32–45 (2015)
12. Eruhimov, V., Meeussen, W.: Outlet detection and pose estimation for robot continuous operation. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 2941–2946. Institute of Electrical and Electronics Engineers (IEEE) (dec 2011)
13. Fan, Q., Brown, L., Smith, J.: A closer look at Faster R-CNN for vehicle detection. In: IEEE Intelligent Vehicles Symposium, Proceedings. pp. 124–129. Institute of Electrical and Electronics Engineers Inc. (aug 2016)
14. Gao, T., Packer, B., Koller, D.: A segmentation-aware object detection model with occlusion handling. In: CVPR 2011. pp. 1361–1368. IEEE (2011)
15. Hamledari, H., McCabe, B., Davari, S.: Automated computer vision-based detection of components of under-construction indoor partitions. Automation in Construction **74**, 78–94 (feb 2017)
16. Hosang, J., Benenson, R., Dollár, P., Schiele, B.: What makes for effective detection proposals? Tech. rep., Max Planck Institute for Informatics (2015)
17. Huang, J., You, S.: Detecting objects in scene point cloud: A combinational approach. In: 2013 International Conference on 3D Vision. pp. 175–182 (2013)
18. Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S., Murphy, K.: Speed/accuracy trade-offs for modern convolutional object detectors. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7310–7311. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach (2017)
19. Kim, P., Chen, J., Cho, Y.K.: Robotic sensing and object recognition from thermal-mapped point clouds. International Journal of Intelligent Robotics and Applications **1**(3), 243–254 (2017)
20. Krispel, U., Evers, H.L., Tamke, M., Viehauser, R., Fellner, D.W.: Automatic Texture and Orthophoto Generation From Registered Panoramic Views. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (2015)
21. Kropp, C., Koch, C., König, M.: Interior construction state recognition with 4d bim registered image sequences. Automation in construction **86**, 11–32 (2018)
22. Lee, M., Kim, H., Paik, J.: Correction of barrel distortion in fisheye lens images using image-based estimation of distortion parameters. IEEE ACCESS **7**, 45723–45733 (2019)
23. Li, J., Liang, X., Wei, Y., Xu, T., Feng, J., Yan, S.: Perceptual generative adversarial networks for small object detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1951–1959 (2017)
24. Liang, Z., Shao, J., Zhang, D., Gao, L.: Small object detection using deep feature pyramid networks. In: Lecture Notes in Computer Science (including subseries

Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). vol. 11166 LNCS, pp. 554–564. Springer Verlag (sep 2018)

25. 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. pp. 740–755. Springer Verlag (2014)
26. Lopes, R., Yin, D., Poole, B., Gilmer, J., Cubuk, E.D.: Improving Robustness Without Sacrificing Accuracy with Patch Gaussian Augmentation. ArXiv (2019)
27. Maalek, R., Lichti, D.D., Ruwanpura, J.Y.: Automatic Recognition of Common Structural Elements from Point Clouds for Automated Progress Monitoring and Dimensional Quality Control in Reinforced Concrete Construction. Remote Sensing **11**(9), 1102 (may 2019)
28. Maji, S., Malik, J.: Object detection using a max-margin Hough transform. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1038–1045. Institute of Electrical and Electronics Engineers (IEEE) (mar 2009)
29. Meeussen, W., Wise, M., Glaser, S., Chitta, S., McGann, C., Mihelich, P., Marder-Eppstein, E., Muja, M., Eruhimov, V., Foote, T., Hsu, J., Rusu, R.B., Marthi, B., Bradski, G., Konolige, K., Gerkey, B., Berger, E.: Autonomous door opening and plugging in with a personal robot. In: Proceedings - IEEE International Conference on Robotics and Automation. pp. 729–736 (2010)
30. Michie, D.: ”memo” functions and machine learning. Nature **218**(5136), 19–22 (1968)
31. Perez, L., Wang, J.: The Effectiveness of Data Augmentation in Image Classification using Deep Learning. ArXiv (dec 2017)
32. Quintana, B., Prieto, S.A., Adán, A., Bosché, F.: Door detection in 3D coloured point clouds of indoor environments. Automation in Construction **85**, 146–166 (jan 2018)
33. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**, 1137–1149 (2015). https://doi.org/10.1109/TPAMI.2016.2577031
34. Ren, Y., Zhu, C., Xiao, S.: Small Object Detection in Optical Remote Sensing Images via Modified Faster R-CNN. Applied Sciences **8**(5), 813 (may 2018)
35. Shorten, C., Khoshgoftaar, T.M.: A survey on image data augmentation for deep learning. Journal of Big Data **6**(1), 60 (2019)
36. Son, H., Kim, C., Kim, C.: Fully Automated As-Built 3D Pipeline Extraction Method from Laser-Scanned Data Based on Curvature Computation. Journal of Computing in Civil Engineering **29**(4) (jul 2015)
37. Song, L., Gong, D., Li, Z., Liu, C., Liu, W.: Occlusion robust face recognition based on mask learning with pairwise differential siamese network. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 773–782 (2019)
38. Turcsany, D., Mouton, A., Breckon, T.P.: Improving feature-based object recognition for X-ray baggage security screening using primed visualwords. In: Proceedings of the IEEE International Conference on Industrial Technology. pp. 1140–1145 (2013)
39. Wu, R., Yan, S., Shan, Y., Dang, Q., Sun, G.: Deep image: Scaling up image recognition. arXiv preprint arXiv:1501.02876 **7**(8) (2015)