Synthetic images generation for semantic understanding in facility management

Luca Rampini and Fulvio Re Cecconi
Department of Architecture, Built Environment and Construction Engineering, Politecnico di Milano, Milan, Italy

Received 7 September 2022
Revised 11 December 2022
20 January 2023
Accepted 31 January 2023

Abstract

Purpose – This study aims to introduce a new methodology for generating synthetic images for facility management purposes. The method starts by leveraging the existing 3D open-source BIM models and using them inside a graphic engine to produce a photorealistic representation of indoor spaces enriched with facility-related objects. The virtual environment creates several images by changing lighting conditions, camera poses or material. Moreover, the created images are labeled and ready to be trained in the model.

Design/methodology/approach – This paper focuses on the challenges characterizing object detection models to enrich digital twins with facility management-related information. The automatic detection of small objects, such as sockets, power plugs, etc., requires big, labeled data sets that are costly and time-consuming to create. This study proposes a solution based on existing 3D BIM models to produce quick and automatically labeled synthetic images.

Findings – The paper presents a conceptual model for creating synthetic images to increase the performance in training object detection models for facility management. The results show that virtually generated images, rather than an alternative to real images, are a powerful tool for integrating existing data sets. In other words, while a base of real images is still needed, introducing synthetic images helps augment the model's performance and robustness in covering different types of objects.

Originality/value – This study introduced the first pipeline for creating synthetic images for facility management. Moreover, this paper validates this pipeline by proposing a case study where the performance of object detection models trained on real data or a combination of real and synthetic images are compared.

Keywords Computer vision, Artificial intelligence, Digital twin, Object detection, Asset management

Paper type Research paper

1. Introduction

The operations and maintenance (O&M) stage represents the most extended phase in the life cycle in the architecture, engineering, construction and operations (AECO) sector

Authors' contribution: LR: conceived of the presented idea, developed the method, performed the computation, contributed to the interpretation of the results; FRC verified the analytical method, contributed to the interpretation of the results. All authors discussed the results and contributed to the final manuscript.

Declaration of competing interest: The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Construction Innovation
Vol. 24 No. 1, 2024
pp. 33-48
Emerald Publishing Limited
1471-4175
DOI 10.1108/CI-09-2022-0232

(Akcamete *et al.*, 2019). During this phase, many stakeholders handle processes and procedures, often appearing and leaving at different times, causing a loss or a distortion of asset information. According to the National Institute of Standards and Technology (NIST), approximately 57.8% of the projected US$15.8bn annual costs are incurred by owners and operators during the operational phase (Gallaher *et al.*, 2004). These expenses are brought on by ineffective business process management, redundant facility management (FM) systems, lost productivity, rework costs and other problems.

Nowadays, a solution for better information management is represented by digital twins (DTs) – an updated and accurate digital replica of a physical asset that represents the asset's as-is condition (Ioannis Brilakis *et al.*, 2019). However, it is necessary to detect objects and their geometric relationships within the asset to generate DTs. Most research is focused on recognizing large architectural components such as columns, ceilings and walls rather than detecting secondary building components like heating, ventilation and air conditioning (HVAC) elements, which are a crucial part of effective FM. However, the production and the update of precise and reliable DTs that reach FM information level present different challenges:

- Compared to the design and construction phases, the operation stage is dynamic, with many changes in uses, tools and pieces of furniture of the various parts that constitute the asset.
- DT is characterized by many objects and systems that are usually smaller than structural elements, making their representability and updatability difficult.
- Compared to structural components, FM-related assets have a wider range of variance within classes, necessitating learning additional feature patterns. For instance, radiators will have slightly varied markings, valve designs and other features.

The significant manual effort required to create an enriched DT is prohibitively expensive compared to the resulting model's perceived value. For these reasons, there is a high demand for greater automation in creating an information-rich DT. Using image processing and machine learning techniques, researchers have recently studied methods for extracting features such as color, texture and shape that can distinguish target components from other objects. In this context, Deep learning models have been applied for contextual awareness of scenes as computers have advanced with the introduction of faster GPUs. Such applications necessitate the collection of a large amount of labeled image data. In this context, large-scale publicly available data sets like the Scene Understanding (SUN) database (Song *et al.*, 2015), Common Objects in Context (COCO) data set (Lin *et al.*, 2014), KITTI Cityscapes data set (Geiger *et al.*, 2012) and NuScenes (Caesar *et al.*, 2019) have been generated. However, data on FM-related scenes are scarce. Thus, there is a vital requirement for large-scale annotated image data regarding the asset's operations components.

The variety of types, shapes and materials of FM-related objects complicates the implementation of a recognition model that performs well. While preparing data for asset components scene understanding, two challenges arise: the first is that image labeling is done by hand, which is time-consuming and expensive. Image labeling for object recognition is the task of using a polygon to mark the area containing the object and specify the class of the object. A dozen clicks on a single object are required to mark a polygon. The second problem is that domain knowledge is necessary to label the FM items. Identifying the area and class of objects in a scene photograph requires expertise. For example, identifying a color code that indicates the contents of a pipe in a plumbing system requires competence. This knowledge may require additional training for the labeler.

These two challenges can be addressed using existing 3D object drawings that are often used inside BIM models. Indeed, in the past few years, many vendors provided detailed and accurate digital representations of FM components. Those virtual models are collected in several open data sets that can be used as a source for generating FM-BIM models. Hence, the labeling operation can be performed in a BIM environment using a virtual camera. However, variances in color and texture between BIM and real-world images – usually photographs – cause differences in spatial elements that serve as training requirements. As a result, for BIM images to be used as training data for photograph analysis, they must first be translated into a photographic style.

Recently, it is increasingly common to use open-source graphics engines, such as Blender, Unity or Unreal, in computer vision and machine learning to generate synthetic data. Synthetic data is information that is artificially manufactured rather than generated by real-world events, which has several significant benefits, including the option to automatically generate labeled data and the possibility to respond to different environmental factors, such as various lighting situations, seasons and day–night cycles.

To date, no synthetic data pipeline has been developed in the context of enriching FM-BIM models. Therefore, this study addresses the following research questions:

RQ1. What pipeline can be used to generate FM-related synthetic data?

RQ2. Are FM synthetic data valuable to increase the accuracy of FM components' object detection?

Finally, the rest of this paper is structured as follows: previous research on the topic and state of the art is presented in Section 2; the proposed and adopted pipeline is described in Section 3; the experiment and results are shown in Section 4; discussion, conclusions and future work are in Section 5.

2. Background

In this paper, we propose a framework based on synthetic data to enable and facilitate the detection of small secondary objects to enrich DT with sufficient details for FM operations. Most of the previous research focused on detecting structural objects such as floors, ceilings and walls (Wang *et al.*, 2017; Hou *et al.*, 2019; Hong *et al.*, 2021; Pan *et al.*, 2022a; Pan *et al.*, 2021), while few studies posed attention to small objects that are parts of assets sub-systems, such as fire safety or energy systems. Compared to structural components, FM-related objects are typically smaller than structural parts with changing geometrical features. Hence, using the same algorithms to detect such small-scale elements is challenging. Therefore, if proven effective, synthetic data can significantly help in training object detection models with a sufficiently diverse and balanced data set without relying on manual collection and annotation of big data sets. In the following paragraphs, we revise object detection applications in the AECO domain and how synthetic data have been introduced and deployed in recent research.

2.1 Object detection models

The goal of generic object detection is to locate and identify existing items in a single image and then label them with rectangular bounding boxes to indicate their certainty of existence. The frameworks of generic object identification algorithms are divided into two groups (Figure 1). One follows the standard object detection pipeline, first generating region proposals and then categorizing each proposal. The other considers object identification a

regression or classification problem, using a unified framework to directly produce final findings (categories and locations).

For this study, we decided to deploy the classification-based method because it can process images in real time, which is beneficial for the surveying activities conducted in FM's scope. Specifically, we trained the You Only Look Once (YOLO) model, introduced by Redmon *et al.* (2015). YOLO uses the topmost feature map to predict confidences and bounding boxes for numerous categories. Since its introduction, it has been constantly improved by adopting several innovative strategies such as batch normalization, anchor boxes, multi-scale training, etc. In particular, we used the fourth version introduced by Bochkovskiy *et al.* (2020). In a nutshell, the input image is divided into a $S \times S$ grid by YOLO, and each grid cell is responsible for guessing the object centered in that grid cell. Each grid cell forecasts B bounding boxes and their associated confidence scores.

Despite the availability of the YOLO v4 model, we cannot use its weights entirely because they have been trained to detect categories that are not included in our application domain. However, we can use a good portion of the already pre-trained weights by applying a transfer learning technique. As its name suggests, transfer learning is the process of applying previously learned knowledge to solve new but related challenges. We can take advantage of a pre-trained model that was previously trained on thousands of images for detecting high-level features as the model has already "seen" and "learned" from many images.

2.2 Detection of secondary objects in buildings

While most of the research is used to assist robots in recognizing certain items in the environment and performing a given task, there has been little effort in the AECO area. Adán *et al.* (2018) suggested a method for detecting objects such as switches, ducts and signs in a colored point cloud. Potential zones of interest are computed through in-depth pictures and color images concerning the wall plane, depending on whether the objects have geometric or color discontinuities in the wall area. The region of interest is then compared to a pre-defined depth model database and a pre-defined color model database containing object classes from the scene. Wei and Akinci (2019) deployed the DenseNet model for feature extraction (Huang *et al.*, 2017) in different publicly available data sets. The authors proposed a framework for image-based localization and semantic understanding that relied on semantic segmentation. However, the conclusion pointed out that object detection models can improve the pipeline mentioned above because only a part of the object is necessary to link it to its DT; therefore, a coarser bounding box might be good enough to enable the association. Finally, (Pan *et al.*, 2022b) proposed a pipeline to enrich geometrical digital twin GDT) by leveraging images and valuable text information. Despite reaching good performances in some object categories, such as fire extinguishers and smoke alarms, for

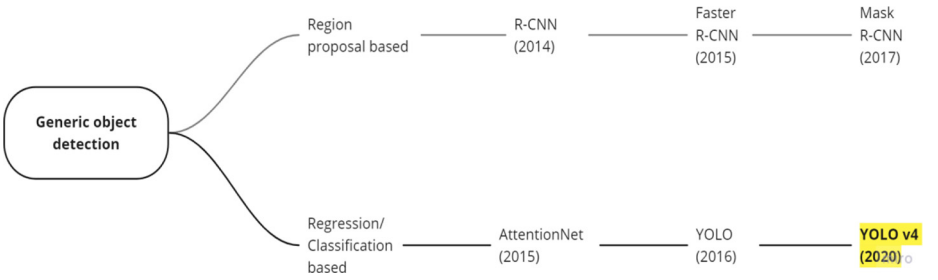

Figure 1.
The two object detection frameworks are region proposal and regression/ classification (derived from Zhao *et al.*, 2018)

objects that vary significantly in different environments (e.g. lights and sockets), the accuracy drops, meaning that the deployed data sets were insufficient to train the model. As a result, the proposed method needed more data to be effective, especially to cover other objects that were not considered in the study but are still an essential component of GDT (e.g. bookshelves, desks, etc.). Our study aims to fill this gap by proposing artificially generated synthetic data.

2.3 Synthetic data

The use of synthetic data to improve the performance of a taught model has become a widespread practice in the computer vision and machine learning communities (Rampini and Cecconi, 2022). Because training a deep learning model generally necessitates a huge quantity of data, many academics have proposed using synthetic data to complement current data sets and offer training data for new applications. The challenge of deploying synthetic data is bridging the reality gap with real-world data. However, the expenses in terms of time and computational power required to generate a good amount of photorealistic data negate the primary selling point of artificially generated data, which is the possibility of generating a large amount of already labeled data essentially for free (Tremblay *et al.*, 2018). Therefore, recent approaches focused on creating diverse scenarios by changing objects' 3D models (Peng *et al.*, 2015) and backgrounds (Saleh *et al.*, 2018).

In the AECO domain, there is scarce research on synthetic data, and none of them is focused on FM-related objects. Hong *et al.* (2021) proposed a pipeline for automatically generating labeled and high-quality synthetic data. The research focused on structural elements from buildings and bridges and comprised three main steps:

(1) converting BIM images into real-world images using CycleGAN;

(2) automatically labeling them with the help of the spatial data in the BIM to produce different synthetic data sets; and

(3) combining the final synthetic data set created by splicing the chosen synthetic data sets.

Other research built synthetic data sets for different purposes: Neuhausen *et al.* (2020) enhance the performance of a YOLOv3 detector in tracking and monitoring workers' movements on the constructions site by adding around 600 synthetically generated in eight construction on-site scenes; Sutjaritvorakul *et al.* (2020) deployed a fully synthetic data set to detect worker from a load-view crane camera contributing to the safety of crane's operation. Moreover, recent research showed the potential of synthetically generated indoor scenes, where many images with different furniture and light conditions were provided with high photorealistic footage (Li *et al.*, 2019; Roberts *et al.*, 2020; Neuhausen *et al.*, 2020; Sutjaritvorakul *et al.*, 2020; Li *et al.*, 2019; Roberts *et al.*, 2020). Finally, Wei and Akinci (2021) proposed a pipeline for generating synthetic data using 4D-BIM for scene understanding. In particular, the fourth dimension of BIM is leveraged to deal with the dynamic environment that is usually characterized on-site construction field. However, to date, no workflows focus on synthetic images representing FM secondary objects; hence, this study aims to fill this gap.

3. Proposed solution

This research is part of a broader schema that automatically enriches DT models with FM-related objects (Figure 2). Aside from those relatively significant structural aspects, smaller

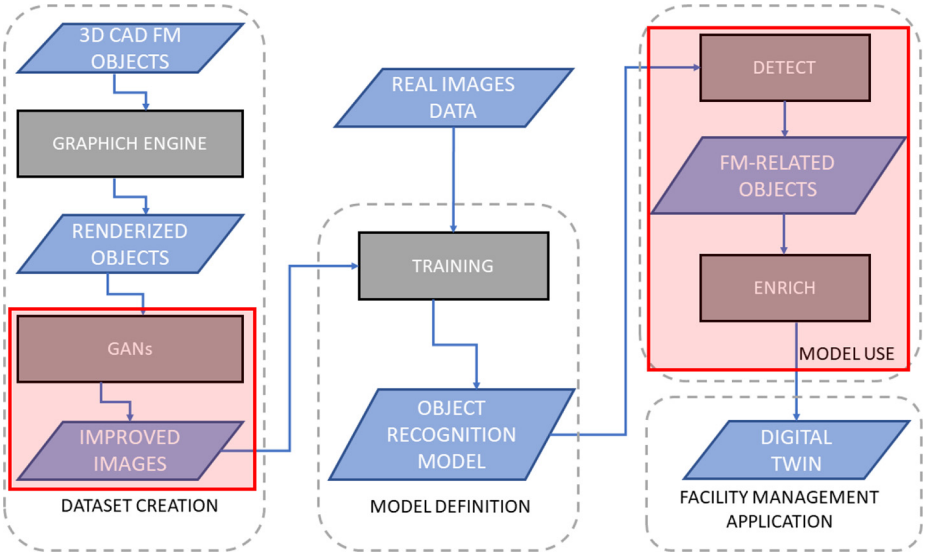

Figure 2.
The overall research schema (the parts colored in red are not covered in this paper)

FM items (such as fire alarms and emergency switches) should be included in an enhanced DT to assist facilities management. Indeed, HVAC systems expenditures often account for the majority of overall costs in a building's operational activities (Adán *et al.*, 2018). As a result, a DT would be more valuable if it included components that are commonly required in FM activities.

The pipeline proposed in this study covers most of the data set creation and model definition phases. The model use, which includes the geometric relationships among the objects, is left outside the scope of this study, and it is partially addressed in Pan *et al.* (2022b).

The workflow contains two primary steps:

(1) the procedural generation of synthetic images through a graphic engine (Blender) and 3D CAD models; and

(2) the training of an object recognition model made by a combination of real and synthetic images. In the following subparagraphs, the two steps are furtherly explained.

3.1 Synthetic data set

Over the past decade, deep learning models have achieved impressive performance and evolved from simple classification tasks to more complex topics such as object detection and/or segmentation. For most applications, rather than increasing the model's complexity, the focus has shifted to providing sufficient data to train the model, especially for computer vision tasks.

The availability of an extensive, well-balanced and precisely labeled data set is an ideal situation that is rarely verified in real-world applications. Often, data sets are scarce or require much time and annotation effort. For example, the ImageNet data set, one of the

most important in the Computer Vision field, required almost two years to label around 11 million images with crowdsourcing methods (Russakovsky *et al.*, 2015).

As a result, many researchers are trying to avoid the challenges of collecting and annotating data in the real world by creating virtual environments that generate training examples in a controllable and customizable manner. These artificially generated data are known as synthetic data sets and present several advantages over real-world-based data:

- produce balanced data sets;
- cover a wide range of lighting conditions;
- generate automatically labeled images;
- produce more semantic representations such as depth maps, segmentation maps and so on; and
- it is easier to comply with privacy regulations such as the EU General Data Protection Regulation (GDPR) (Voigt and Bussche, 2017) or the California Consumer Privacy Act (CCPA) (Bukaty, 2019). Especially in a little-shared industry such as construction, it is challenging to collect publicly available indoor images from different buildings.

Figure 3 shows the workflow's difference between a real-world, manually annotated dataset and a synthetically generated one.

Generally, the input data for generating synthetic images is an environment built with 3D assets. In the AECO industry, thanks to the growing adoption of CAD first and BIM lately, a considerable amount of 3D drawings is widely present in the market. Moreover, most of these models are available in open-source data sets formed by models freely provided by vendors. Therefore, it is possible to leverage the existing 3D BIM models and use them as a backbone for generating AECO-related synthetic data sets. In this study, we used the BIM object platform (BIMobject, 2015), which contains different object categories (with different formats) that covers several building systems (structural, electric, heating and so on). From BIMobject, it was possible to download the 3D models and import them into a graphic engine.

A graphic engine uses computer time rather than human time to generate examples. It has complete information about the scenes it renders, allowing it to save time and money on human annotations and reviews. A graphic engine also allows for the generation of rare examples, allowing control over the training data set's distribution. For this study, we used Blender – a free and open-source 3D graphic engine that supports three-dimensional object modeling, simulation and rendering (blender.org, 2015). The engine was chosen among the others for its embedded Python API (Blender Python API), which allows scripts that facilitate the process of iterating through light conditions, camera poses and textures, as well as the process of generating annotations.

3.2 Complete data set

Despite the abovementioned advantages of synthetic data, training and running an object detection model that relies entirely on artificially generated data cannot guarantee good performances. Even if the real-world data are limited to 10% of the entire data set, the benefits in terms of precision and recall are well documented (Nowruzi *et al.*, 2019).

Although there are no existing data sets for object detection focusing on FM-related objects, inside the largest annotated image data set – Google Open Images V6 – we can use the "power plugs and sockets" category to identify those elements. The data set was released in 2020 and comprised 1.9 million images for 16 million manually annotated

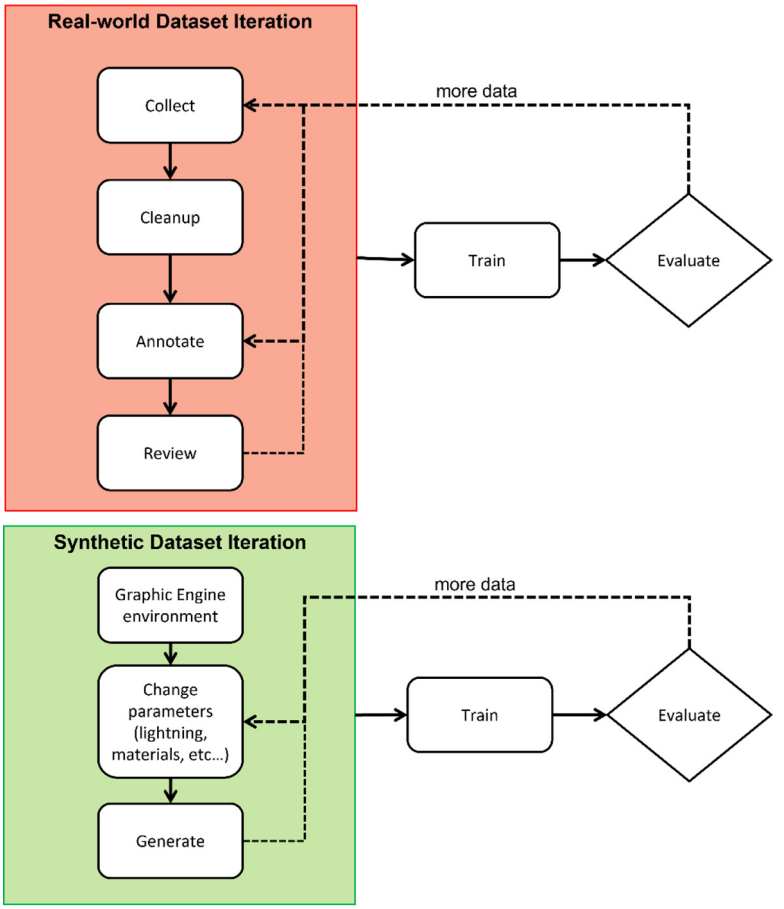

Figure 3.
The dataset iteration process is compared using real-world and synthetic datasets

Notes: *Top*: A real-world dataset necessitates the steps of collect, cleanup, annotate, and review. These steps necessitate costly and time-consuming human labor. *Bottom*: Creating a synthetic dataset requires creating an environment with 3D assets, adjusting randomization parameters, and running the environment to generate new data. The datasets include accurate annotations and are validated automatically, eliminating most time-consuming steps
Source: Derived from Borkman *et al.* (2021)

bounding boxes for 600 object categories. For the "power plugs and sockets" category, 112 images are available for training, totaling 198 bounding boxes.

On the other hand, the pipeline proposed to create synthetic images tailored explicitly for FM applications is shown in Figure 4.

Eevee and Cycles are the two main render engines accessible in Blender. In general, Eevee was designed for real-time rendering, whereas Cycles was designed for realism. As a result, unless dedicated graphics cards (GPUs) are used, Cycles renders images substantially slower than Eevee. Currently, a user can build all accessible forms of ground truth maps when using

Cycles, unlike Eevee, which cannot generate segmentation masks or optical flow ground truths. Mayer *et al.* (2018) presented a thorough examination of several synthetic and real-world data sets for training neural networks for optical flow and disparity estimation applications. The authors highlight two significant findings. The first point is the significance of diversity in the training set. Second, they demonstrate that the photo-realism of the data set is not significantly influencing the model's strong performance. Therefore, to introduce a pipeline as accessible and flexible as possible, we used the Eevee render engine to build the synthetic data set.

The output of the generation process is a series of RGB synthetic images and associated text files that contain the coordinates of the bounding boxes surrounding the sockets. The text files are produced in the format compatible with YOLO, where each object's bounding box inside the picture is reported in the form: object class, x coordinate of the bounding box center, y coordinate of the bounding box center, bounding box width an bounding box height.

Consequently, to test the accuracy of using a combination of real and synthetic images, we created 100 images (like the ones in Figure 5) by rendering and iterating through ten different types of sockets, ten different settings (kitchen, bathroom, bedroom and living room) with different camera poses and light conditions.

3.3 Object detection model

In this step, we aim to use the created data set to test the effectiveness of using artificially generated data to increase precision and recall in detecting small objects related to FM context.

The model implemented for this study, which is the fourth version of the YOLO architecture (Bochkovskiy *et al.*, 2020), has been chosen for two main reasons:

(1) It can detect objects almost in real time using videos and photos: this aspect is significant considering how the surveys are conducted in our field, where the use of wide-angle cameras and drones is growing dramatically.

(2) Despite the newer version of YOLO (up to v7), the fourth version is more robust and has been used and proven in several applications. However, changing the version of the model in the future should not change too much the pipeline steps proposed in this study.

3.4 Evaluation metrics

To test the effectiveness of introducing synthetic data, we must define the metrics we use to evaluate the model. In object detection tasks, the predictions are made using a bounding box

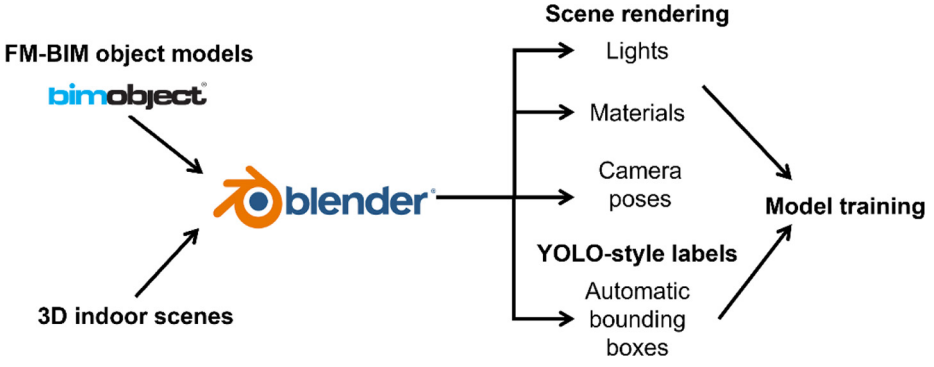

Figure 4.
Proposed pipeline for creating FM-related synthetic images

Figure 5.
Examples of synthetically generated data

and a class label. The overlap between the predicted and the ground truth bounding boxes determines the accuracy of the prediction, which is commonly called intersection over union (IoU) (Figure 6).

A threshold for the IoU value needs to be defined to calculate the precision and recall of the object detection model. For instance, if the IoU threshold is 0.5, and the IoU value for a prediction is 0.8, then it is considered a true positive (TP). On the other hand, if the IoU is 0.4, the prediction is classified as false positive (FP). Therefore, by setting different IoU thresholds, the prediction's precision and recall differ. Usually, the precision–recall curve is adopted to represent the tradeoff between precision and recall for different thresholds. A high area under the curve indicates both strong recall and high precision, with high precision corresponding to a low false positive rate and high recall corresponding to a low false negative rate.

The average precision (AP) is defined by the area under the precision–recall curve. For multi-classes detection tasks, the most common metric is the mean AP (mAP) score, defined below:

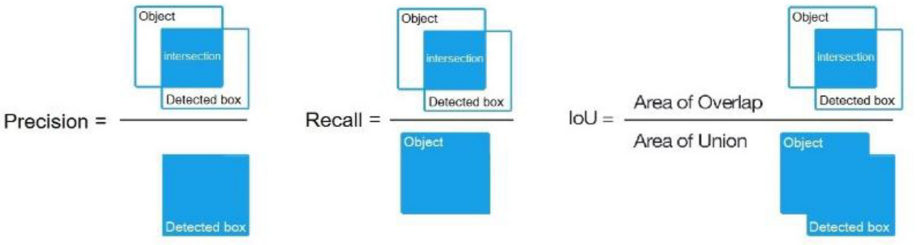

Figure 6.
Calculation process of precision, recall and IoU

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i$$

where N is the number of classes.

As we are detecting one class in this study, the AP is equivalent to the mAP.

We evaluated the mAP for four different cases, summarized in Table 1.

The first data set includes only real-world images and is used as the benchmark. The second data set is the same size as Data set 1, and it is used to understand how the introduction of synthetic images alters the model's performance. Although Data set 2 presents an unlikely practical scenario (i.e. where synthetic images are used to replace real images to maintain the data set size), this is useful to understand how the performance change when introducing synthetic images without increasing the data set size. Finally, Data sets 3 and 4 are mixed data sets, i.e. composed of real and virtually generated images, where the amount of synthetic images is 50 and 90% of the total images, respectively.

4. Results and discussions

In this section, we evaluate the performances of the YOLO v4 object detection model with the four data sets. We use the mAP, a common evaluation metric for object detection tasks. The comparison clarifies if the proposed pipeline can be used for FM-related tasks, perhaps extending the methodology to other components such as fire extinguishers, furniture or heating equipment. Finally, we discuss the results and the insights that can be derived from them.

4.1 Object detection results

The training of the object detection model has been performed using the NVIDIA Tesla P100 graphic card with 16 GB of vRAM. The model requires as input 416 × 416 RGB images, which are divided into batches of 32 images. The number of epochs for training is set to 2,000 after trying different values (more epochs were causing overfitting, were less a drop in performance), and the learning rate was set to 0.001.

Figure 7 shows the mAP performance of the validation set during the training. The validation set is formed by 20% of the real images training set, except for Data set 4, where the validation set is 40% of the real images (i.e. 5% of the overall training data set). We decided to use only real images for the validation set because also the test data set included them. In this way, the performances on the validation set are comparable to those on the test set. In all data sets, the models are increasing the mAP value until it reaches a stable value with the epoch increase, meaning that the model has converged.

Moreover, Table 2 shows the mAP performances on the test data set. Noteworthy, the mAPs on the test and validation data sets are similar, assessing the robustness of model performance.

The worst performance is in the case of the smallest data set (Data set 2), composed equally of real and synthetic images. Considering that a data set composed of only real images (Data set 1) performs better, it follows that, for the same size, data sets formed of only real images perform better. However, it should be remembered that the primary

Data set	Real images	Synthetic images
1	112	0
2	66	66
3	112	100
4	112	888

Table 1. Data sets used to train the object detection model

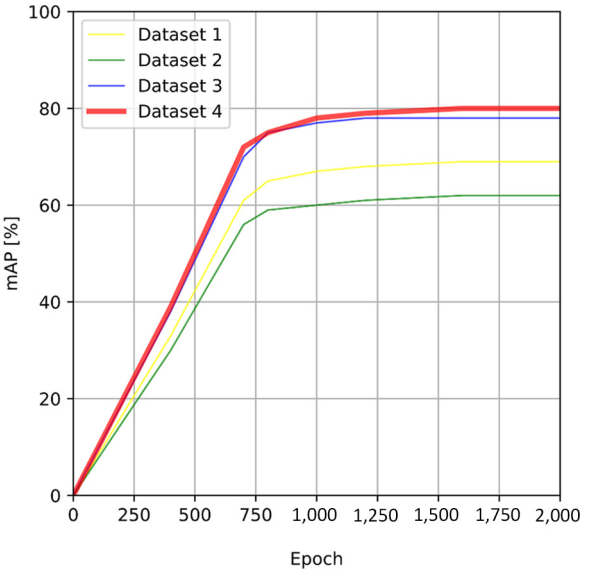

Figure 7.
mAP performance on the validation set for each data set. The model is learning most of its weights in the first 1,000 epochs and reaches a stable value in the last epochs

Data set	mAP (%)
1	*69*
2	*61*
3	*77*
4	*79*

Table 2.
mAP performances of the three data sets

purpose of synthetic images is to enable model training when sufficient real images are unavailable. Therefore, it is more interesting the results obtained with Data sets 3 and 4 compared to Data set 1. In this case, there is an increase in performance with both Data sets 3 and 4.

It should be noted that there is no significant difference between Data set 3 (composed of 50% synthetic images) and Data set 4 (composed of 90% synthetic images). Therefore, we can infer that the benefits of adding synthetic images are most significant when the number of images added is comparable with the initial data set. By contrast, the benefits are significantly less when adding more images than the original data set.

The results are also affected by the distance between the cameras and the target object. As sockets and power plugs are small objects, the training images were taken not too far from the objects (around 50 cm). Therefore, introducing pictures of an entire room may not trigger the detection of the targeted objects. However, the speed of the YOLO network (close to real time) allows for adjusting the distance quickly by giving immediate feedback to the user (especially with video cameras).

4.2 Discussions

The previous paragraph shows that the introduced pipeline can be deployed to enhance the object detection model's performance by leveraging synthetically generated data. The

experiments, aiming at recognizing sockets and power plugs, gave numerous insights and reflection points.

First, for the same number of images, the data set composed of only real images performed better than the mixed data set (real and synthetic), meaning that the virtually generated images are a powerful tool for enhancing and integrating existing real image-based data sets rather than replacing them. This is evident from Data sets 3 and 4, where the mAP score increased by a factor of 10% compared to Data set 1. Therefore, synthetic images are a viable solution to address the challenges that FM-related object detection models face for effective training. Moreover, this performance gap might be reduced by addressing the sim-to-real gap with other AI techniques, such as generative adversarial networks (GANs). Second, the possibility to easily create annotated synthetic data allowed us to embrace the FM object's variability. For instance, in some scenarios like the ones in Figure 8, the model trained on Data set 1 could not correctly predict sockets and power plugs because neither black nor horizontally oriented sockets were present in the training data set. Solving these problems using only real-world images would have required collecting and annotating images that include the cases above, spending considerable time and resources. On the other hand, synthetic images were sufficient to change color or orientation to the 3D BIM models already used. Moreover, creating images capable of recognizing these types of scenarios also took very little time. This means that synthetic data can easily solve problems related to the variability of FM objects, perhaps iterating the training process once problems are found through feedback and testing.

Finally, it is worth mentioning that the performance of Data sets 3 and 4 are comparable, meaning that many synthetic images do not guarantee a comparable increase in performance. Consequently, it is hard to establish a minimum required number of images to achieve the desired results because they depend on different aspects like object size, typologies, etc. Therefore, defining the correct number of synthetic images to include will be established by an iterative process (somewhat like what happens when defining the depth and density of layers of neural networks).

In conclusion, synthetic images facilitate the introduction of more data quickly and enable iteration throughout the process. If the performance is not good enough, more images can be created to train the model again without conducting another campaign for data collection and annotation.

Figure 8.
Examples of socket types not included in Data set 1 and, therefore, not recognized by the model. By adding synthetic images based on 3D drawings similar to the power plugs in the picture, the model can recognize them

5. Conclusion

The documentation of as-is conditions has increasingly been done by capturing videos and images. However, to automate the process of extracting information from such data, it is necessary to train deep learning models that require producing a large amount of labeled data, which is a costly and timely demanding task. In this study, we introduced a pipeline tailored for the AECO industry for generating FM-related synthetic images to overcome difficulties with classifying ground truth visual FM data. Although other methodologies have been proposed for different industries, the proposed method takes advantage of the existing 3D BIM object models that are freely accessible to create a training data set that encompasses the broadest possible collection of FM-related objects. Moreover, using a graphic engine allows the production of more realistic images by deploying advanced rendering tools that help to close the sim to the real gap. The methodology has been tested to recognize sockets and power plugs to answer the first research question. However, the proposed method can produce the desired amount of virtually generated data of any objects modeled in a BIM environment using only open-source and freely available sources and products.

The created data set has been used to train a YOLO object detection model, and its performances are compared with those obtained using real training data. As an answer to the second research question, the experiment findings demonstrated that the suggested strategy outperforms models trained on only real-world photos by covering a broader object's variability and increasing prediction robustness. In the future, we plan to introduce a GAN model to further increase the realism of the synthetic images and probably improve the mAP score and the variability of the scenes.

References

Adán, A., Quintana, B., Prieto, S.A. and Bosché, F. (2018), "Scan-to-BIM for 'secondary' building components", *Advanced Engineering Informatics*, Vol. 37, pp. 119-138, doi: 10.1016/J.AEI.2018.05.001.

Akcamete, A., Akinci, B. and Garrett, J.H. (2019), "Potential utilization of building information models for planning maintenance activities", *in EG-ICE 2010 - 17th International Workshop on Intelligent Computing in Engineering*.

BIMobject (2015), "BIM objects", available at: www.bimobject.com/en (accessed 8 August 2022).

blender.org (2015), " Blender.org, blender.Org", available at: www.blender.org/ (accessed 8 August 2022).

Bochkovskiy, A., Wang, C.-Y. and Liao, H.-Y.M. (2020), "YOLOv4: optimal speed and accuracy of object detection", doi: 10.48550/arxiv.2004.10934.

Borkman, S., Crespi, A., Dhakad, S., Ganguly, S., Hogins, J., Jhang, Y.C., Kamalzadeh, M., Li, B., Leal, S., Parisi, P. and Romero, C. (2021), "Unity perception: generate synthetic data for computer vision", doi: 10.48550/arxiv.2107.04259.

Bukaty, P. (2019), *The CA Consumer Privacy Act (CCPA): an Implementation Guide/Preston Bukaty*, The CA Consumer Privacy Act (CCPA): an implementation guide, IT Governance Publishing. Ely, Cambridgeshire, UK.

Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G. and Beijbom, O. (2019), "'nuScenes: a multimodal dataset for autonomous driving", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 11618-11628. doi: 10.48550/arxiv.1903.11027.

Gallaher, M.P., *et al.* (2004), "Cost analysis of inadequate interoperability in the U.S. Capital facilities industry", Nist [Preprint].

Geiger, A., Lenz, P. and Urtasun, R. (2012), "Are we ready for autonomous driving? The KITTI vision benchmark suite", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3354-3361. doi: 10.1109/CVPR.2012.6248074.

Hong, Y., *et al.* (2021), "Synthetic data generation using building information models", *Automation in Construction*, Vol. 130, p. 103871, doi: 10.1016/j.autcon.2021.103871.

Hou, X., Zeng, Y. and Xue, J. (2019), "Detecting structural components of building engineering based on deep-learning method", *Journal of Construction Engineering and Management*, Vol. 146 No. 2, p. 4019097, doi: 10.1061/(ASCE)CO.1943-7862.0001751.

Huang, G., Liu, Z., Van Der Maaten, L. and Weinberger, K.Q. (2017), "Densely connected convolutional networks", *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition*, CVPR 2017, 2017-January, pp. 2261-2269. doi: 10.1109/CVPR.2017.243.

Ioannis Brilakis, A., Pan, Y., Borrmann, A., Mayer, H.G., Rhein, F., Vos, C., Pettinato, E. and Wagner, S. (2019), "Built environment digital twining", available at: https://aspace.repository.cam.ac.uk/handle/1810/318329 (accessed 4 August 2022).

Li, W., Saeedi, S., McCormac, J., Clark, R., Tzoumanikas, D., Ye, Q., Huang, Y., Tang, R. and Leutenegger, S. (2019), "Interiornet: mega-scale multi-sensor photo-realistic indoor scenes dataset", *in British Machine Vision Conference 2018, BMVC 2018*, available at: https://interiornetdataset.github.io. (accessed 9 August 2022).

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C.L. (2014) 'Microsoft COCO: common objects in 'context, *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8693 LNCS(PART 5), pp. 740-755. doi: 10.48550/arxiv.1405.0312.

Mayer, N., Ilg, E., Fischer, P., Hazirbas, C., Cremers, D., Dosovitskiy, A. and Brox, T. (2018), "What makes good synthetic training data for learning disparity and optical flow estimation?", *International Journal of Computer Vision*, Vol. 126 No. 9, pp. 942-960, doi: 10.1007/s11263-018-1082-6.

Neuhausen, M., Herbers, P. and König, M. (2020), "Using synthetic data to improve and evaluate the tracking performance of construction workers on site", *Applied Sciences (Switzerland)*, Vol. 10 No. 14, doi: 10.3390/app10144948.

Nowruzi, F.E., *et al.* (2019), "How much real data do we actually need: analyzing object detection performance using synthetic and real data", doi: 10.48550/arxiv.1907.07061.

Pan, Y., Braun, A., Borrmann, A. and Brilakis, I. (2021), "Void-growing: a novel scan-to-BIM method for Manhattan world buildings from point cloud".

Pan, Y., Braun, A., Borrmann, A. and Brilakis, I. (2022a), "3D deep-learning-enhanced void-growing approach in creating geometric digital twins of buildings", *Proceedings of the Institution of Civil Engineers - Smart Infrastructure and Construction*, Vol. 40, pp. 1-17.

Pan, Y., Braun, A., Brilakis, I. and Borrmann, A. (2022b), "Enriching geometric digital twins of buildings with small objects by fusing laser scanning and AI-based image recognition", *Automation in Construction*, Vol. 140, p. 104375, doi: 10.1016/J.AUTCON.2022.104375.

Peng, X., *et al.* (2015), "Learning deep object detectors from 3D models", *Proceedings of the, IEEE International Conference on Computer Vision*, pp. 1278-1286. doi: 10.1109/ICCV.2015.151.

Rampini, L. and Cecconi, F.R. (2022), "Artificial intelligence in construction asset management: a review of present status, challenges and future opportunities", *Journal of Information Technology in Construction*, Vol. 27 No. 43, pp. 884-913, doi: 10.36680/J.ITCON.2022.043. available at: www.itcon.org/2022/43,

Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2015), "'You only look once: Unified, Real-Time object detection", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-December, pp. 779-788. doi: 10.48550/arxiv.1506.02640.

Roberts, M., Ramapuram, J., Ranjan, A., Kumar, A., Bautista, M.A., Paczan, N., Webb, R. and Susskind, J.M. (2020), "Hypersim: a photorealistic synthetic dataset for holistic indoor scene understanding", pp. 10892-10902. doi: 10.48550/arxiv.2011.02523.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. and Berg, A.C. (2015), "ImageNet large scale visual recognition challenge", *International Journal of Computer Vision*, Vol. 115 No. 3, pp. 211-252, doi: 10.1007/S11263-015-0816-Y/FIGURES/16.

Saleh, F.S., *et al.* (2018), '. "Effective use of synthetic data for urban scene semantic 'segmentation", *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pp. 86-103. doi: 10.1007/978-3-030-01216-8_6.

Song, S., Lichtenberg, S.P. and Xiao, J. (2015), "SUN RGB-D: a RGB-D scene understanding benchmark suite", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 07-12-June-2015, pp. 567-576. doi: 10.1109/CVPR.2015.7298655.

Sutjaritvorakul, T., Vierling, A. and Berns, K. (2020), "Data-driven worker detection from load-view crane camera", in *Proceedings of the 37th International Symposium on Automation and Robotics in Construction, ISARC 2020: From Demonstration to Practical Use – To New Stage of Construction Robot. International Association on Automation and Robotics in Construction (IAARC)*, pp. 864-871. doi: 10.22260/isarc2020/0119.

Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., To, T., Cameracci, E., Boochoon, S. and Birchfield, S. (2018), "Training deep networks with synthetic data: bridging the reality gap by domain randomization", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2018-June, pp. 1082-1090. doi: 10.48550/arxiv.1804.06516.

Voigt, P. and Bussche, A. V D (2017), "The EU general data protection regulation (GDPR): a practical guide", p. 385.

Wang, R., Xie, L. and Chen, D. (2017), "Modeling indoor spaces using decomposition and reconstruction of structural elements", *Photogrammetric Engineering and Remote Sensing*, Vol. 83 No. 12, pp. 827-841, doi: 10.14358/PERS.83.12.827.

Wei, Y. and Akinci, B. (2019), "A vision and learning-based indoor localization and semantic mapping framework for facility operations and management", *Automation in Construction*, Vol. 107, p. 102915, doi: 10.1016/J.AUTCON.2019.102915.

Wei, Y. and Akinci, B. (2021), "Synthetic image data generation for semantic understanding in everchanging scenes using BIM and unreal engine", *Computing in Civil Engineering 2021 – Selected Papers from the ASCE International Conference on Computing in Civil Engineering 2021*, pp. 934-941. doi: 10.1061/9780784483893.115.

Zhao, Z.Q., *et al.* (2018), "Object detection with deep learning: a review", *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 30 No. 11, pp. 3212-3232, doi: 10.48550/arxiv.1807.05511.

Further reading

Blender Python API (2022) Blender 3.2 Python API Documentation — Blender Python API (2023), available at: https://docs.blender.org/api/current/ (accessed 8 August 2022).

Corresponding author
Luca Rampini can be contacted at: luca.rampini@polimi.it