Machine Learning Identification of Building Services

Michael Biddle

Mark Reynolds
School of Mathematics, Physics and Computing
University of Western Australia

Scott Renner-Hahn
CEED Client: Ezametric

Abstract

This project involves the development of an object detection model which uses machine learning techniques including the You Only Look Once (YOLO) algorithm to identify building services within an image. Once developed, the object detection model is intended to be integrated into Ezametric's products to improve surveying techniques for construction industry designers. A model for identifying pipe bends was built using a training data set of 483 images, where an independent testing set of 50 images indicated a precision of 96% and a recall of 75% for the model. In addition, separate segmentation techniques were investigated for cable tray identification, however the tested segmentation techniques resulted in low identification accuracies due to the images containing complex texture boundaries. Any factors arising from the investigations that may influence the output accuracy and performance have also been documented.

1. Introduction

Building services engineers undergo laborious surveying tasks for refurbishment projects, involving lots of ladder movement, safety considerations, and guesswork in regions of limited access such as ceiling voids. Surveying a building properly takes patience and expertise, where if critical information is missed, it can cause weeks of project delays or suboptimal solutions, lowering client and end-user satisfaction (Renner-Hahn, 2022).

Globally, engineers in the industry tend to rely on readily available tools and equipment such as pens, paper and cameras for documenting existing sites. This process can be slow and tedious, and in limited access situations can sometimes be impossible. To help improve this process, Ezametric has proposed a modelling software which identifies building services in a set of images and creates models that record the location and size of each object.

The identification of building services is required as a component of this approach. Techniques for identifying objects in images typically involves machine learning tools and algorithms which will be researched and developed for this project.

1.1 Literature Review

1.1.1 Convolutional Neural Network Models

The essence of object detection is to summarise object sizes and locations in rectangular bounding boxes and then classify them to train a software model in identifying classes from new images (Xiao et al., 2020). Object detection is typically conducted via convolutional neural networks (CNNs) which perform numerous layers of mathematical equations on existing labelled data. CNNs are very accurate, but require significant amounts of data to produce the intended results. Extensive time and training are required for the approach to be applicable in everyday applications (Redmon et al., 2016). The use of a one-stage CNN, like the Single Shot Multi-Box (SSD) or the You Only Look Once (YOLO) detectors, is typically for applications where results are required in real time (Xiao et al., 2020).

1.1.2 "You Only Look Once" Detector

The You Only Look Once (YOLO) detector is a state-of-the-art, real time object detection algorithm introduced in 2015. The algorithm detection process is very similar to the SSD, but establishes object detection as a regression problem for spatially separating bounding boxes and associated class probabilities (Redmon et al., 2016). YOLO is the leader in object detection due to its speed, detection accuracy, generalisation ability, and choice of open-source driven improvements (Keita, 2022).

The newest version of YOLO is YOLOv7 which improves its architectural level for use in more diverse features by integrating extended efficient layer aggregation networks. It also implements a trained "bag of freebies" that increases the model's detection accuracy and speed without increasing training costs (Wang et al., 2022). As this framework incorporates the improvements made over the course of all YOLO versions, it is considered the best current object detection model, and thus is the algorithm of choice for this project.

1.1.3 Model Evaluation Techniques

The utilised model evaluation techniques for YOLOv7 include precision and recall. The precision of a model measures the accuracy of a model by finding the proportion of correct predictions amongst all true positives (TP) and false positives (FP) (Hui, 2019) True positives include all correctly identified objects and false positives involve all identifications that are incorrectly mistaken for the intended object.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

The recall of a model measures the sensitivity of the model by calculating how well the positives are found in comparison to false negatives (FN) (Hui, 2019). False negatives wrongly indicate that a particular condition is absent and thus include any objects missed from detection.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

1.2 Project Objectives

The aim of this project is to design an object detection model which utilises machine learning tools and algorithms for identifying building services within an image. According to the priorities of the current industry, pipe bends and cable trays are among the most important objects for identification. The following objectives break down this project's aim.

1. Complete image collection and data processing for complete data sets of images.
2. Design, train and test the YOLOv7 Object Detection Model for pipe bends and adapt the model for use with other building services objects.
3. Investigate parameters that impact the performance and accuracy of the model.

2. Methodology

The methodology for the design and development of an object detection model required for this project can be split into five distinct phases: Image collection, image processing, YOLOv7 training, YOLOv7 testing, and bounding box segmentation.

2.1 Software Tools and Resources

The software tools utilised for this project include python algorithms which are mostly accessed online or self-designed. Several python scripts have been created throughout the development of the object detection model and are collated in a GitHub repository.

The most specialised resource is a High-Performance Computer (HPC) with Graphics Processing Units (GPUs), required for bulk machine learning training and testing. UWA has provided "Kaya," the university's HPC, for use on the project.

2.2 Design/Development Approach

2.2.1 Image Collection

Multiple sets of images of building services are required for both the training and testing data sets. These images can be collected manually with a camera or extracted through online sources such as Google Images. In most cases, the number of images extracted online was not sufficient for a complete training set, and thus several images were needed to be captured manually. Recording the quality of these images was also important for investigating any impacts on the detection model's accuracy and performance.

Two python scripts were created to assist with the online image collection process. The first script, "download_images.py", takes an input of Google searches and automatically downloads a given number of images into a directory. The second script, "filter_images.py", filters through the downloaded images, removing duplicates and unreadable images.

2.2.2 Image Processing

Development of the training data set involves drawing bounding boxes around each of the objects that the detection model will be trained to identify. The program "LabelImg" was used to create bounding boxes and record the coordinates of the corners of each box in a text file. The bounding boxes are required for all images in the training data set making the image processing phase slow and tedious.

2.2.3 YOLOv7 Training

The training phase is automated by the python script, "training_gpu.py", which utilises the YOLOv7 package to setup and conduct the training process with the required input conditions and training set of images and labels. The training process records the precision and recall of the model as it trains and validates itself through multiple evolutions of the data set. Once completed, the model produces its highest accuracy evolution condensed into a python script which can be used for testing on a separate set of images.

2.2.4 YOLOv7 Testing

The testing phase is automated by the python script, "testing_gpu.py", which utilises the output script of the training phase and a data set of testing images to form predictive bounding boxes around the objects which it believes are the intended building services of interest. The results of the predictive bounding boxes are then manually assessed on accuracy based on true positives, false positives, and false negatives. To confirm the precision and recall, roughly 10% of the total collected images are set aside for testing.

2.2.5 Bounding Box Segmentation

To locate the object more precisely within its generated bounding box, a Factorization-Based Segmentation (FSEG) technique, can be applied on the bounding box area to separate the space into regions of different textures. As the object occupies majority of the bounding box space, the most prominent texture is isolated forming a 'mask' around the exact shape of the object which will be implemented into the overall program's model.

Figure 1 displays a flow diagram for the outputs of a test pipe bend image as it passes through the YOLOv7 testing phase and the bounding box segmentation phase.

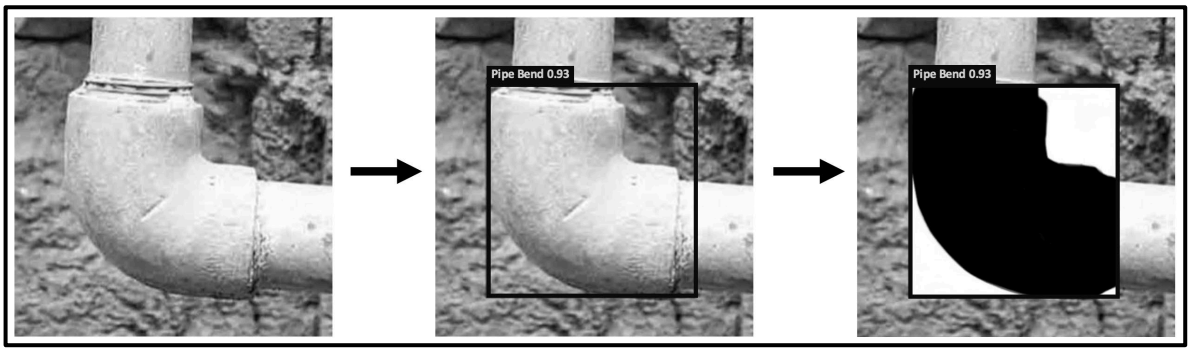

Figure 1 Flow diagram of object detection model outputs.

3. Results and Discussion

3.1 Pipe Bend Image Collection and Processing

A total of 3500 images (500 each for 7 unique Google searches) of pipe bends were downloaded from online sources. Of these, 504 unique and readable images were considered for manual filtering, and only 52 images were deemed suitable from the total dataset. A further 481 images were manually captured in various locations around Perth. Of the resultant set of 533 images, 483 were assigned to training and 50 were assigned to testing where test images were selected based on investigating all potential scenarios.

3.2 Pipe Bend Training Results

The training process elapsed 7.64 hours to complete training of 100 generations of the data set and produce its most optimal object detection script. The model converged to an overall self-validated precision of 94.1% and recall of 76% upon completion.

3.3 Pipe Bend Testing Results

After manual observations of the test results across 50 images, a total of 135 pipe bends were detected with 5 false positives and 44 false negatives. This results in a precision of 96% and a recall of 75%. It is important to note that many of the false negatives included pipe bends hidden in the background of an image or obstructed by other objects, with one highly condensed image of lower quality containing 15 false negatives itself.

3.4 Segmentation Observations for Cable Trays

Since cable trays often span an entire image, it is difficult to apply a bounding box approach to the object without inadvertently highlighting the entire image. To combat this complication, a segmentation approach was taken where textures within an image are isolated and identified. Figure 2 displays the produced texture patterns of a selected cable tray image for 2 segmentation techniques: Binarized Entropy and FSEG.

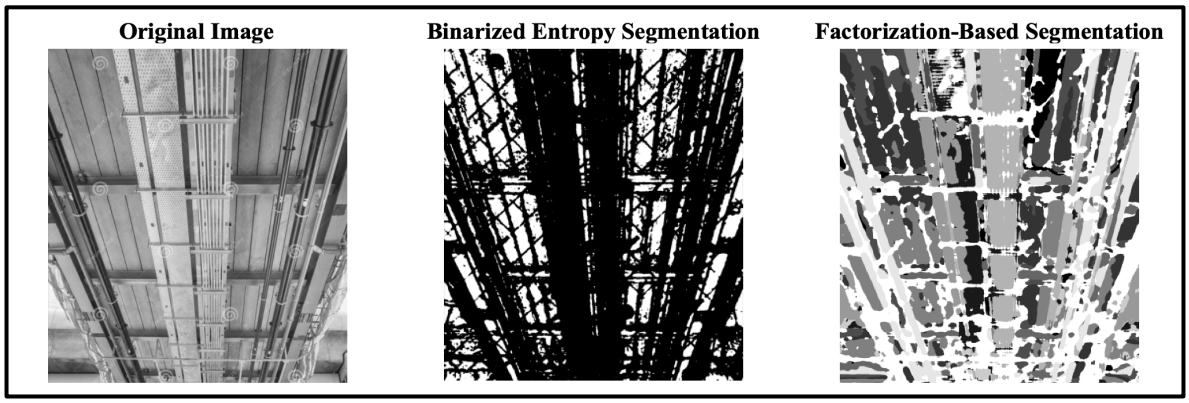

Figure 2 Segmentation texture patterns for a cable tray image.

This approach led to an inaccurate identification of cable trays due to the images often being too complex with uncertain texture boundaries.

4. Conclusions and Future Work

Overall, the results of pipe bend training and testing produced promising data which will be useful for integrating into Ezametric's software. A precision in the high 90% range means that we can be very confident that a box predicted by the model contains the intended object and is not mistaken for a different object. A moderately-high recall percentage around 75% means that any pipe bend within an image is likely to be detected by the model, with the most prominent pipe bends being more likely for detection than those in the background.

The remainder of this project will be spent on conducting segmentation techniques on bounding box areas to finalise the process mentioned in section 2.2.5.

As Ezametric requires the identification of several building services for a complete identification model, the training and testing process will need to be repeated for each of the desired objects. This can be completed using the YOLOv7 detection model designed throughout this project. However, additional research into alternative techniques is required for objects which span across images or have undefined length.

5. Acknowledgements

The author would like to thank academic supervisor, Mark Reynolds, and client mentor, Scott Renner-Hahn, for all their help throughout the project. Many thanks will also extend to UWA's HPC Manager, Chris Bording, for allowing use of UWA's hardware resources.

6. References

Hui, J. (2019, April 3). *mAP (mean Average Precision) for Object Detection*. https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173

Keita, Z. (2022, September). *YOLO Object Detection Explained: A Beginner's Guide*. Datacamp. https://www.datacamp.com/blog/yolo-object-detection-explained

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, *1*(1). https://doi.org/10.1109/cvpr.2016.91

Renner-Hahn, S. (2022, October). Ezametric Business Plan for Investors and Grants.

Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. https://arxiv.org/pdf/2207.02696.pdf

Xiao, Y., Tian, Z., Yu, J., Zhang, Y., Liu, S., Du, S., & Lan, X. (2020). A review of object detection based on deep learning. *Multimedia Tools and Applications*, *79*(33), 23729–23791. https://doi.org/10.1007/s11042-020-08976-6