Machine Learning Tool Development in Fire Safety Design Review

Min-Ching CHIOU¹,*, Wen LIU² and Kelvin WONG³

¹Arup, 37/F, Huaihai Plaza 1045, Huaihai Road (M), Xuhui District, Shanghai 200031, China
²,³Arup, Room 3008, 30/F, Jing Guang Centre Hu Jia Lou, Chaoyang District, Beijing 100020, China

*Corresponding author

Keywords: Object detection, Architectural drawings, Fire code compliance assessment.

Abstract. The feasibility of object detection technique to recognize and localize the less-featured building elements on the architectural drawings was tested. An object detection engine using Faster R-CNN deep learning techniques was trained that machine can learn from existing architectural drawings what the different variations of the building components look like. Then machine is capable to tag these elements on a 2D architectural drawing, and engineers can apply some rule-based checks to flag up any design that violates the building code of practice. It is proved that machine can go through a large amount of design drawings quickly, recognize where the important building components are located, and report whether the design conforms to the building codes.

Introduction

In 2016, it was declared that we have entered the era of artificial intelligence. Based on the continuous development of AI/ML technologies, particularly the improvement of GPU and data analysis methodologies, AI engineers to help the human engineers for more efficient and innovative engineering design can be realized in the near future.

AI/ML has the benefit for large throughput computing cases. In fire engineering, the process to review the fire safety design is also a kind of input-check-output work. When the plan layout drawings are received, the fire engineers must check whether all the fire safety design, such as the fire compartment, the number and distribution of safety exits, the travel distance for occupants and etc., are compliant with code requirements or not. If some alterations are needed, the feedback, calculation, suggestions should be provided to clients in a short time. Normally this “hin und zurück” process repeats many times and the optimal solution cannot be immediately proposed at the first stage.

In this research, the possibility and methodologies to implement machine learning technology in fire safety design review will be investigated and developed. As the machine learning engine becomes better over time, the tool can automate repetitive work with a growing confidence. This will allow engineers to concentrate on design optimization in the digital world.

Analytical Approach

Machine Learning

The first part is to develop an object detection process for architectural drawings. Based on the computer vision technique of object detection, building components on an architectural drawing should be identified and properly labelled, and its bounding box should be calculated to indicate its position and orientation. This information should be delivered as the output of this step.

The machine learning workflow for object detection includes:

i) Problem formulation: define and prioritize what classes of objects to be identified, and also what properties of them to be extracted. Since the fire code review stage will be mainly focused on the compliance of egress distance design, machine should be trained to classify at least the egress doors and egress stairs on architectural drawings.

ii) Algorithm research: explore proper object detection methods and their restriction.
Creating the deep learning pipeline: define where the DL module lives and what the structure of the neural network is.

Data processing:
- Data collection: collect standard architectural drawings within Arup.
- Data labelling: prepare the image data with labelled object bounding boxes.

Define evaluation metrics: average precision (AP), the standard procedures laid out in the PASCAL VOC Challenge, will be used to evaluate how good the object detection algorithm is performing.

Model training and testing: split the dataset into training, validation and testing; first two for model training and last one is to verify that the accuracy is sufficient. The output should be a list of building components identified in an image and their respective bounding boxes.

**Engineering Application**

After the position of doors, egress stairs on an architectural drawing can be localized via machine learning tool, the next stage is to use detected position information for further geometric computation via Python coding to review the code compliance of egress distance.

The engineering application workflow for egress distance review includes:

i) Find external boundary: find the external contour of the building layout.
ii) Import bounding boxes: read the bounding box locations of safety exits from object detection results.
iii) Calculate pixel distances: save the pixel coordinates of each point on the image and then calculate the pixel distance between any point inside the external building contour and the nearest safety exit.
iv) Convert to physical distance: compute the ratio between pixel distance and physical distance and then convert all the pixel distances calculated from previous step to real physical distance.
v) Check with fire code: assign colours for different physical distance to indicate the egress distance for every point inside the building to the nearest safety exit.

**Machine Learning Analytics Methodology Implementation**

**Object Detection Algorithm Selection**

Object detection is a technique to train the machine localize and classify the objects inside an image. This task typically includes proposing region of interest (bounding box proposal) and then classify the objects inside them. It can be expressed as follows,

\[
\text{(Multiple) Object Detection} = \text{Object Location (bounding box)} + \text{Object Classification (feature extraction)}
\]

Along with the proliferation of powerful GPUs and availability of large datasets, object detection algorithm based on the deep neural networks has become state-of-the-art nowadays. For those deep neural networks/deep learning, Convolutional Neural Network (CNN) performs well and has been proven very effective in applications of image recognition and classification [1,2].

Most deep-learning-based object detection approaches repurpose image classifiers by applying them to a sliding window across an input image. Some approaches such as R-CNN (Region-based Convolutional Neural Network) make region proposals using selective search instead of doing an exhaustive search to save computation [3]. In this study, Faster R-CNN is applied for the object detection to find out building elements and their positions on architectural drawings [4].

**Analytics Environment Establishment**

The analytics environment for this study includes:
- Operating System in Linux: Ubuntu 16.04
- GPU: NVIDIA Titan Xp Pascal
- Parallel Computing Platform and API: Cuda 8.0
Dataset Preparation

The training data for object detection should include the bounding box and label of each object on the image. LabelImg [5], a graphical image annotation tool, is used in this study to prepare the labelled images with annotation file saved as an xml file in the PASCAL VOC 2007 format [6].

Results

Machine Training and Testing: Door Detection

Two object detection tests for door elements are conducted in this study. Table 1 summarized the setup of deep learning model training process for door detection.

<table>
<thead>
<tr>
<th>Test no.</th>
<th>test01</th>
<th>test02</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objective of the test</td>
<td>detect the precise region of door on a small image</td>
<td>detect multiple objects, e.g. different type of doors, on the architectural drawings of dimension around 1000×1000 pixels</td>
</tr>
<tr>
<td>Dimension of dataset image</td>
<td>(200<del>500)×(200</del>500) pixels</td>
<td>1000×(600~1000) pixels</td>
</tr>
<tr>
<td>No. of dataset image</td>
<td>858</td>
<td>3500</td>
</tr>
<tr>
<td>numClass</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>[train, val, test] images</td>
<td>[430, 214, 214]</td>
<td>[2800, 400, 300]</td>
</tr>
<tr>
<td>Iteration times</td>
<td>[10000, 5000, 10000, 5000]</td>
<td>[10000, 5000, 10000, 5000]</td>
</tr>
<tr>
<td>Network Model</td>
<td>VGG16</td>
<td>VGG16</td>
</tr>
</tbody>
</table>

Some test results for door detection on small image (test01) are shown in Fig. 1. Most of the door elements on the images can be detected where the average precision of door class reaches 0.85; however, there are still some bounding boxes with incorrect, imprecise or missing error types.

A key finding from test01 is a double door could be partially detected as a single door if all kinds of doors are labelled as one class. Since the position of whole double door is required for the engineering application stage, dataset preparation for door object in test02 are categorized into 9 door object classes. Fig. 2 shows the test results to detect the doors in the core of an office building in the image of 1000×1000 pixels based on the trained test02 model. It demonstrated that the trained multiple object detection model can classify and localize different types of doors appearing in the image.

Machine Training and Testing: Stair Detection

The setup of deep learning model training for egress stair detection is shown in Table 2.

<table>
<thead>
<tr>
<th>Test no.</th>
<th>test03</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objective of the test</td>
<td>detect the precise region of egress stair on a small image</td>
</tr>
<tr>
<td>Dimension of dataset image</td>
<td>(200<del>500)×(200</del>500) pixels</td>
</tr>
<tr>
<td>No. of dataset image</td>
<td>314</td>
</tr>
<tr>
<td>numClass</td>
<td>1</td>
</tr>
<tr>
<td>[train, val, test] images</td>
<td>[158, 78, 78]</td>
</tr>
<tr>
<td>Iteration times</td>
<td>[80000, 40000, 80000, 40000]</td>
</tr>
<tr>
<td>Network Model</td>
<td>VGG16</td>
</tr>
</tbody>
</table>

Fig. 3 shows some test results for egress stair detection on small image (test03). The average precision of egressstair class is 0.91 even though the training dataset of stair is less. Most testing results of stair detection are as the type 1 in Fig. 3 that the location and precise region of egress stair can be detected; however, there are still some bounding boxes with incorrect or imprecise error types.
Engineering Application: Egress Distance Review

In this study, the python codes to review of egress distance requirement in China fire code is developed following the process in section 2.2. The demonstration sample is a standard plan layout of an open space office in high-rise building. The linear egress distance from any point inside the building to the nearest safety exit, which means the entrance door of vestibule for smoke-proof staircase, should not be more than 37.5 meters.
The outcome is shown in Fig. 4, where the blue colour is linear egress distance less than 10m; the light blue colour is linear egress distance between 20m and 10m; the green colour is linear egress distance between 30m and 20m; the yellow colour is linear egress distance between 37.5m and 30m; linear egress distance > 37.5m will be visualized in red colour.

Figure 4. Visualization of egress distance measurement.

Conclusion
In this study, the preliminary tests of Faster R-CNN object detection method are successfully conducted to train the machine classify and localize different types of building elements appearing on architectural drawings as the database for further engineering application such as egress distance code review automation. It can be expected that machine can browse and understand building layout via computer vision techniques, and afterwards help engineers to facilitate the code review process.

To have further engineering applications on fire code review, more building elements, such as walls, rooms, and corridors, should be detected by machine. This may require some other computer vision techniques and therefore, will need further investigation and examination in the next development stage.

Acknowledgement
This research was financially supported by the Invest In Arup research funding.

References
[5] https://github.com/tzutalin/labelImg