

# Architectural Drawing Recognition

## *A case study for training the learning algorithm with architectural plan and section drawing images*

Can Uzun<sup>1</sup>, Meryem Birgül Çolakoğlu<sup>2</sup>

<sup>1,2</sup>Istanbul Technical University

<sup>1,2</sup>{uzunc|colakoglumer}@itu.edu.tr

*This paper aims to develop a case study for training an algorithm to recognize architectural drawings. In order to succeed that, the algorithm is trained with labeled pixel-based, architectural drawing (plan and section) dataset. During the training process, transfer learning (pre-training model) is applied. The supervised learning and convolutional neural network are utilized. After certain iterations, the algorithm builds awareness and can classify pixel-based plan and section drawings. When the algorithm is shown a section that is not produced with conventional drawing technic but through hybrid technics, it could predict the drawing class correctly with %80 of accuracy. On the other hand, some of the algorithm prediction is misoriented. We examined this prediction problem in the discussion section. The results illustrate that neural networks are successful in training algorithms to recognize and classify pixel-based architectural drawings. But for a highly accurate algorithm prediction, the dataset of the drawing images must be ordered, according to sample resolution, sample size and sample coherence for the dataset.*

**Keywords:** *Classification Algorithm, Pixel-Based Architectural Drawing Recognition, Plan, Section*

### INTRODUCTION

In this paper, we used convolutional neural networks to build awareness of architectural drawings for algorithms. We outlined the framework for reading pixel-based architectural drawings. We utilized transfer learning method to make training more robust.

In recent years, the interests have been increasing on the training of the algorithm with drawing datasets (Kaiyrbekov & Sezgin, 2019, Ha & Eck, 2017). These works generally attend to make the algorithm

generate and recognize stroke-based vector drawing. There have been very few works done related to architectural drawing recognition and generation process. In 2018, Huan & Zheng represented an architectural drawing recognizer and this study is one of the precedents of architectural drawing recognition field. Hua & Zheng (2018), used a dataset that consists of only plan drawings. This study differs from other studies with its dataset. This dataset is based on pixel-based plan and section drawings. And the aim

is the recognition of the algorithm of pixel-based architectural drawings.

One of the most important limitations for the recognition of pixel-based architectural drawing is the lack of publicly available neat and tidy drawing datasets. So the generation of an architectural drawing dataset is crucial for deeplearning field.

## BACKGROUND

In machine learning, classification is a method that sorts the elements of a dataset into labeled categories (Kesevaraj and Sukumuran 2013). The learning algorithm that can solve the classification problems is called classifier. Classifiers are primarily used for image recognition tasks. The identity of objects, shapes, and faces are recognized by learning algorithms through classification method. Image recognition tasks are utilized primarily in healthcare, automobile, security, and retail industries [1]. Face recognition, autonomous vehicles are some of the output of the image recognition tasks. In this study, we propose an image recognition task through which the trained algorithm can detect the architectural drawing type.

The problem space of an architectural drawing recognition task differs from pervasive image recognition tasks. Most of the recognition tasks use a dataset that has realistic photographs. A dataset that contains architectural drawings, consists of lines. Lu & Tran (2017) explains the differences between a drawing dataset and a dataset that contains photographic real images: A dataset that consists of realistic photographs can be created through the web easily. The drawing data that consists of various style is limited in number. Drawing dataset has either less complex grayscale images, while photographic image dataset is more complex and has different values on RGB channels for every pixel. Drawing dataset consists of void space information between lines but photographic images consist of visual information (Lu & Tran, 2017). Because of these reasons, the dataset which consists of images of drawings has low feature. In deep learning low feature corresponds to

low dimensionality and is a problem for the training process. An algorithm that is trained with the low featured data overfits. In other words, the algorithm memorizes all the data but not be able to recognize any data that is outside of the dataset. This causes the disadvantage for the drawing classifiers.

There are many studies on the topic of drawing recognition (Ha & Eck, 2017; Lu & Tran 2017; Yesilbek & Zengin, 2017; Xu et al., 2018). These studies focus on the topic of how to train an algorithm to recognize freehand drawings. Hua & Zheng (2018), represents the architectural drawing recognition and generation tasks. Hua & Zheng (2018), used a dataset that contains only plan drawings.

## CASE STUDY: ARCHITECTURAL PLAN AND SECTION DRAWING CLASSIFIER

In this case study, we trained the algorithm with the pixel-based architectural drawings to make an algorithm read and recognize the architectural drawing images.

### Methodology

In machine learning studies, three types of training methods are used: supervised learning, unsupervised learning & reinforcement learning. Here, we use supervised learning method. In Supervised learning, algorithms are fed with a labeled dataset, and compare their output with given labeled data to make predictions (figure 1).

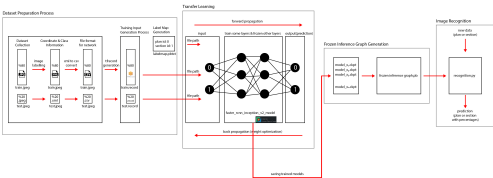


Figure 1  
Algorithm Training  
Process

**Dataset Preparation Process.** The data set contains .jpeg format images that consist of pixel-based plans and sections drawings. The total number of .jpeg pixel-based drawings is two hundred. The data set consists of one hundred pixel-based section draw-

ings and one hundred pixel-based plan drawings. Images were searched via Google images and Fatkun-Batch, batch download plug-in was used to download the pixel-based drawing dataset. After the images were collected, they were manually checked for the repetitive images. Then, the data set was divided into two groups with random selections, for the training and the test datasets. The training dataset consists of 80 plan drawing images and 80 section drawing images. The remaining images were used to create the test dataset. The test dataset is 20% of the whole dataset and consists of 40 drawing images that contain 20 plan and 20 section pixel-based drawings.

This set of .jpeg file format images is not readable by the algorithm. The machine reads the matrices formed by the position and RGB information in the pixels. Before training the algorithm, the image labels that are the identities of the data must also be defined. So labeling that is a manual process was carried out to explain to the algorithm the identities of the drawing class. Labellmg software was used for labeling (figure 2).

Figure 2  
Labeling Plans and  
Sections in the Data  
Set



As a result of labeling, each image in the data set has been converted to the .xml extension file format that contains the pixel coordinates and class identity of the data. All the files with the extension .xml in the data set had been converted to .csv (comma separated value) file format to make the machine be able to read the dataset. For both plan and section data, images converted to .xml and .csv formats. The last phase of the dataset process is to generate the .tfrecord files for both training data and test data. Tfrecord file format is necessary to be able to feed the

data to the neural networks. So we converted the .csv data to .tfrecord files. To match the class id with the labeled dataset, we created the label map that contains the plan id as 0, section id as 1. In this way, the output of the neural network prediction can be 0 or 1 denoting the architectural drawing class.

**Training Procedure.** The transfer learning method is utilized for algorithm training. In this study we used the Tensorflow object detection API model (URL2). For some fields, creating enough amount and the proper dataset is hard and sometimes impossible. And accuracy of the machine learning is related to the quality and the amount of the dataset. Transfer learning that is a pre-trained algorithm can be a solution for this type of issues (Zuo, H., et al., 2019). There is no enough amount of architectural plan and section drawing images which is publicly available. So our dataset is not sufficient for training the algorithm in an architectural drawing recognition task. Thus, the previously trained model can be used for a new classification problem. The pre-trained model, Faster-RCNN-Inception-V2-COCO was downloaded from Tensorflow object detection API github page and it was used to classify the plan and section drawings. The tfrecord files for test and training dataset are fed to the pre-trained algorithm with the label maps of the drawing classes.

We terminated the algorithm training process in 1937 iteration that took about 2.5 hours. During the training, loss values were checked from Tensorboard. The loss value that is the error rate of the algorithm prediction is the value to control the training accuracy of the algorithm. Tensorboard is an interface in which the loss values can be controlled in real-time. The training process was stopped when training slowed down. At the end of the training process, observations showed that the lowest loss value was about 0.02 (figure 3).

**Constructing the Image Classifier.** During the training process, checkpoints were created in which the model was registered. Checkpoints provide access to the registered status of trained models. In order to make this model available for drawing recognition

task, checkpoint with the lowest loss value in the iteration is selected and then converted to frozen inference graph. The classifier model uses the frozen inference graph for prediction of the drawing class.

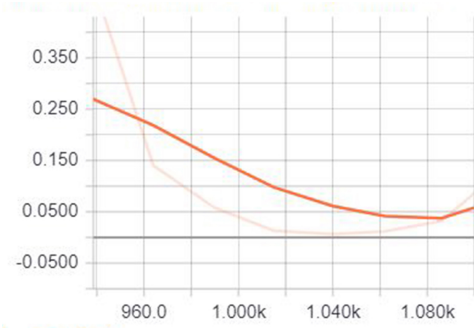


Figure 3  
The Status of the  
Loss Function in  
1110. Iteration on  
the Tensorboard

### Testing and Evaluating of the Drawing Classifier

When we observed the prediction process of the algorithm, high variance and low bias appeared as the problem of this case study. In the high variance and low bias problem, some prediction of the algorithm might be correct but some of them might be disoriented. This means the algorithm predictions are not stable. Some of the predictions of our classifier model are disoriented. This may be the result of overfitting. Overfitting is a common problem in deep-learning. If an algorithm overfits, it means that the algorithm just memorized the dataset but it did not learn the dataset. For our model, we can say that most of the predictions of the classifier are correct. Moreover, when the model predicts the drawing class correctly, the prediction accuracy is high enough (figure 4). This must be the advantage of transfer learning.

In addition to the correct creation of the steps of the algorithm in the machine training, the hardware power is also very effective. In this pilot study, the model was developed on CPU, and so the training process was quite slow. Therefore, the number of iterations in the training of the model had been kept low. During the training, 100% of the CPU was used

and the training slowed down. In addition to the lack of hardware, the fact that the data set is limited to 200 images has significantly reduced the efficiency of the training. Therefore, the development of the data set was considered as one of the most important issue.

As a result of examining the loss values in the Tensorboard during the training, loss diagrams showed unstable values (figure 5). This result may be related to insufficient quantity and quality of the data set.

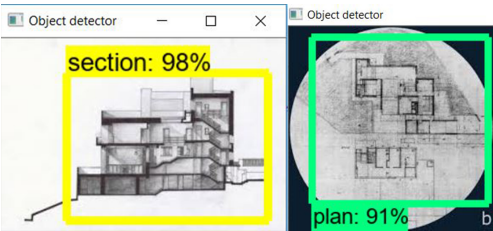


Figure 4  
Plan & Section  
Recognition with  
the Accuracy  
Percentage

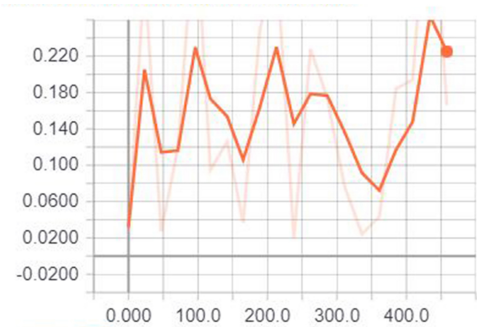


Figure 5  
Unstable Loss  
Values

The sections and plans which were used in the training process of the algorithm, consist of architectural technical drawing images. To understand the effectiveness of the algorithm prediction, we tested the algorithm with silhouette sections which have no technical drawings. These silhouette sections were accurately estimated as sections by the trained algorithm too. But the drawing boundary of the silhouette section is estimated incorrectly (figure 6). Nevertheless, it is a positive result that the algorithm can perceive

the silhouette section as a section. With this prediction of the algorithm, we can say that the algorithm can learn the logic and the structure of the section drawing.

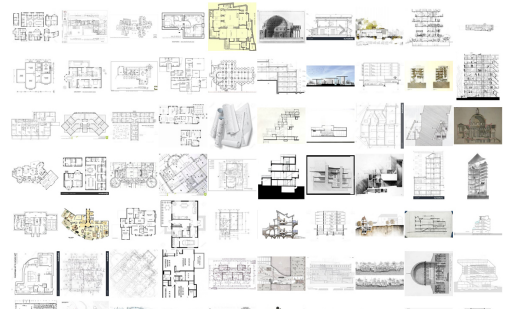
Figure 6  
Silhouette Section  
Recognition with  
the Accuracy  
Percentage



Figure 7  
Some Part of the  
Dataset

### ***The Sample Types in The Dataset:***

The sample types in our dataset vary, so our dataset is heterogeneous (figure 7). And when the dataset is too heterogeneous, the size of the dataset must be more. But our dataset size is limited and, there are some differentiated samples in the dataset. RGB values and resolutions of the samples vary among the data. Because of these reasons, we observed that too much variation in small size dataset affects the training efficiency negatively.



## **DISCUSSION**

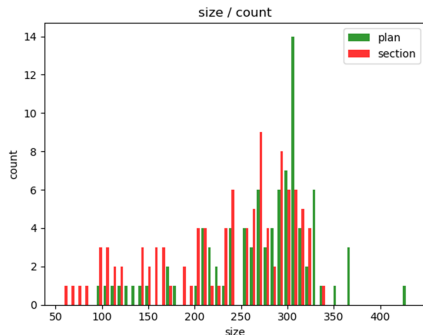
The algorithm, which was tested as a result of the training, had made some incorrect predictions and in many estimations, it incorrectly defined the boundaries of the drawings. The reasons for this situation: the insufficient sample count in the data set, too much heterogeneous data and the varying resolutions of the samples in the dataset.

### ***The Size of The Dataset:***

The data set for the training is not large enough for the classifier algorithm to make correct generalizations. Some deep-learning methods which can make predictions with very little data, are developed (Triantafyllou et al., 2019). But for this study, pervasive deep learning method is applied. Conventional training methods need lots of data. Domingos (2015) & Norvig et. al. (2009) emphasize the importance of the sample amount in a data set to train an algorithm. Deep learning needs thousands of data to make a highly accurate prediction in classification tasks. In our case study, the sample size is limited to just two hundred images. This limitation might decrease some of the accuracies of the algorithm's predictions.

### ***The Resolution of the Samples:***

We examined the resolution of the samples that are fed to the algorithm and saw that resolution for both plan and section drawing images vary. The resolution difference in the dataset makes irregular the sample qualities. And because of this reason, some of the samples may be outliers. Outliers in a dataset decrease the effectiveness of the training session. So for the future study, resolution of the samples in the dataset should be arranged. We can see the resolution values of each sample in the dataset in figure 8.



## CONCLUSION

This paper presents a case study on training the algorithm for pixel-based architectural drawing recognition. The dataset consists of pixel-based (.jpeg) plan and section drawings were collected for the algorithm training process and converted to .xml, .csv, .tfrecord file formats to feed the dataset to the algorithm. The dataset is a small sized dataset that contains 100 plan and 100 section drawing images. In the training process, we used transfer learning for a robust training process. For transfer learning, we used the Tensorflow object detection API model.

Even if the data set is small in size, we have observed that the algorithm with the transfer learning method can make accurate predictions. Some of the predictions were incorrect. This is the reason for the high variance low bias problem. And this problem is related to the dataset size and quality. When we examined our dataset, we observed that the dataset is too heterogeneous, has various resolution values for each data and the size of the data set is too small. Results show that the most important topic for a pixel-based architectural drawing recognition tasks is neat, tidy and enough amount of data. On the other hand, when the model predicts the class of the drawing data, the accuracies are high enough. Although our dataset consists of only architectural technical drawing images, when the algorithm reads a silhouette section, it can predict the drawing class correctly.

This means that the trained algorithm can learn the logic and the structure of architectural drawings.

## REFERENCES

- Domingos, Pedro 2015, *The Master Algorithm. How the Quest for the Ultimate Learning Machine will Remake Our World*, Basic Books, New York
- Ha, David and Eck, Douglas 2017 'A Neural Representation of Sketch Drawings', *arXiv preprint*, p. arXiv:1704.03477
- Halevy, Alon and Norvig, Peter 2009 'The Unreasonable Effectiveness of Data', *IEEE Computer Society*
- Huang, Weixin and Zheng, Hao 2018 'Architectural Drawings Recognition and Generation through Machine Learning', *the 38th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA2018)*, Mexico City
- Kesavaraj G K, G and Sukumaran, Surya 2013 'A study on classification techniques in data mining', *2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*
- Kaiyrbekov, Kurmanbek and Sezgin, Metin 2019 'Stroke-based sketched symbol reconstruction and segmentation', *arXiv preprint*, p. arXiv:1901.03427
- Lu, Wayne and Tran, Elizabeth 2017 'Free-hand Sketch Recognition Classification', *CS 231N Project Report*
- Triantafyllou, Eleni, Zhu, Tyler, Dumoulin, Vincent, Lamblin, Pascal, Xu, Kelvin, Goroshin, Ross, Gelada, Carlos, Swersky, Kevin, Manzagol, Pierre-Antoine and Larochelle, Hugo 2009 'Meta-dataset: A dataset of datasets for learning to learn from few examples', *arXiv preprint*
- Xu, Peng, Huang, Yongye, Yuan, Tongtong, Pang, Kaiyue, Song, Yi-Zhe, Xiang, Tao, Hospedales, Timothy M., Ma, Zhanyu and Guo, Jun 2018 'SketchMate: Deep Hashing for Million-Scale Human Sketch Retrieval', *arXiv preprint*
- Yesilbek, Kemal Tugrul and Sezgin, T. Metin 2017 'Sketch Recognition with Few Examples', *Computers & Graphics*
- Zuo, Huo, Lu, Jie, Zhang, Guangquan and Liu, Feng 2018, 'Fuzzy Transfer Learning Using an Infinite Gaussian Mixture Model and Active Learning', *IEEE Transactions on Fuzzy Systems*, 27 (DOI: 10.1109/TFUZZ.2018.2857725), pp. 291-303
- [1] <https://imagga.com/blog/the-top-5-uses-of-image-recognition/>
- [2] [https://github.com/tensorflow/models/tree/master/research/object\\_detection](https://github.com/tensorflow/models/tree/master/research/object_detection)

Figure 8  
Resolution Values  
of both pixel-based  
Plan and Section  
Drawings