

Architectural Factors Detection from Plan by Deep Learning Framework

Jinho Park¹, Doyoung Yoon² and Woo-Hyoung Lee³

¹Global School of Media, Soongsil University, Seoul, 156-743, South Korea

²Global School of Media, Soongsil University, Seoul, 156-743, South Korea

³Department of Architecture, Namseoul University, Cheonan,
331-707, South Korea

³wlee@nsu.ac.kr

Abstract

This study presents the basics of emotion - based technology system by detecting various elements of architectural drawings as the first step of objectification of aesthetic characteristic and design tendency reflecting architect 's design intention by using example - based machine learning. With the development of technology, new software has been introduced in the field of architectural design, which is able to solve the concerns of environmental values and solve problems, and generalize environmental analysis. However, the aesthetic value compatible with it does not meet the needs of the same context due to social recognition and technical limitations. The purpose of this study is to quantify the scope of subjective judgment of architectural design and to objectify it. In other words, the intention of the designer is the process of confirming the inherent attributes of the result of the expressed architectural design.

Keywords: *aesthetic value, deep learning, object detection, convolutional neural net*

1. Introduction

Urban Modern architecture has developed an aesthetic view of architectural style reflecting cultural diversity based on existing climate and environmental response. These architectural typologies reflect cultural characteristics, diversification of social uses, and socio-economic changes. In addition to the advancement of modern technology, the introduction of new technologies and software in the field of architectural design suggests alternatives to address the concerns and problems of environmental values. However, the aesthetic value that is compatible with it does not meet the needs of the same context due to social recognition and technical limitations. Compared to design automation and systematization of aesthetic values through technological innovation in other design fields are studied in various ways such as modern detection systems R-CNN [1], YOLO (You Only Look Once) [2], and Convolutional Neural Networks (CNN) [3], there is still a desperate situation in the field of architecture [4].

What is clear from the advent of deep learning is that the HOG-based classifier is replaced by a more accurate convolutional neural network-based classifier. But there is one problem. CNN systems are too slow and take a lot of computation time. It is impossible to perform while sliding the window detector on CNN. The R-CNN series solves this problem by using an object proposal algorithm called Selective Search, and most of the object detection algorithms in recent years are based on R-CNN. Selective Search uses local clues such as texture, intensity, and color to create possible positions of objects.

Received (October 26, 2017), Review Result (January 5, 2018), Accepted (January 16, 2018)

The purpose of this study is to identify the various elements of architectural drawings as the first step of objectification of the aesthetic characteristics and design tendencies reflecting the architect's design philosophy.

2. Background

Analyzing the work of a specific architect helps to provide a better understanding of his architectural composition techniques and suggests more advanced methods of architectural design based on this understanding. Therefore, we can find many cases of applying the extracted principle and through the analysis of the architectural works in actuality. It is not just a model to imitate the works of the existing architects but also to establish the principles and theories that appear in the architecture. It is applied to modern architecture by correcting, supplementing and reconstructing.

The aesthetic characteristic of architect is a complex and comprehensive way of building elements. The aesthetic characteristic is determined by the method, of architects' personal selections and responses to independent and interconnected factors such as the relationship with the site, the layout of the building, the formative features, the interior and exterior materials, the plan and section configuration, the connection of internal and external spaces.

Typical examples of distinguishing these aesthetic features are Richard Meier and Mario Botta. They are the architects of modern America and Europe, and they have developed in common with modern architecture based on their own architecture. In the works of the two architects, common efforts to solve all the problems of the space are shown in a common way, and the distinctive differences caused by the ideas and design principles of the influential architects are found. Both architects have commonalities and have developed contrasting architectural characteristics, although they have the same background as those educated by Le Corbusier. In order to distinguish these characteristics, it is required to analyze architecture as a composite of comprehensive factors.

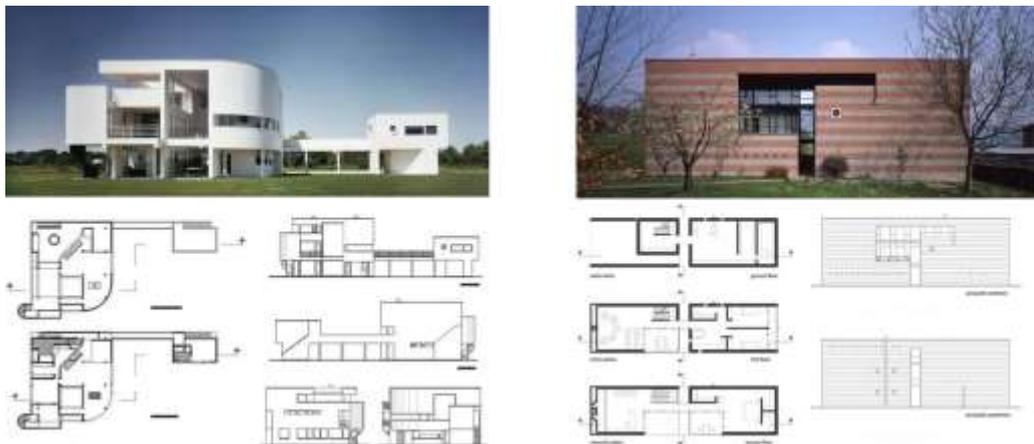


Figure 1. Example of Different Characteristic in Architecture. Left: Saltzman House by Richard Meier Right: Ligornetto House by Mario Botta

In general, the most important analysis attribute in analyzing the aesthetic characteristics of an architect is the spatial structure of a building. Convex Map and Justified Graph method are used as typical research methods to derive this from traditional research methods. The Convex Map defines the spatial space of the object to be analyzed and interprets the topological relationship among individual unit spaces divided into convex spaces. The Justified Graph can clearly see the

connections between spaces by identifying circles as spaces and lines as spatial relationship. Also, deep and shallow structure shapes and mutual tightness can be expressed and their characteristics can be grasped. The purpose of this study is to analyze the aesthetic characteristics of architects by using deep running rather than the traditional spatial structure analysis. In this paper, we try to distinguish the aesthetic characteristics based on the characteristics of individual space unit and connection method of each architect by letting the space unit on the drawing be recognized automatically by machine learning.

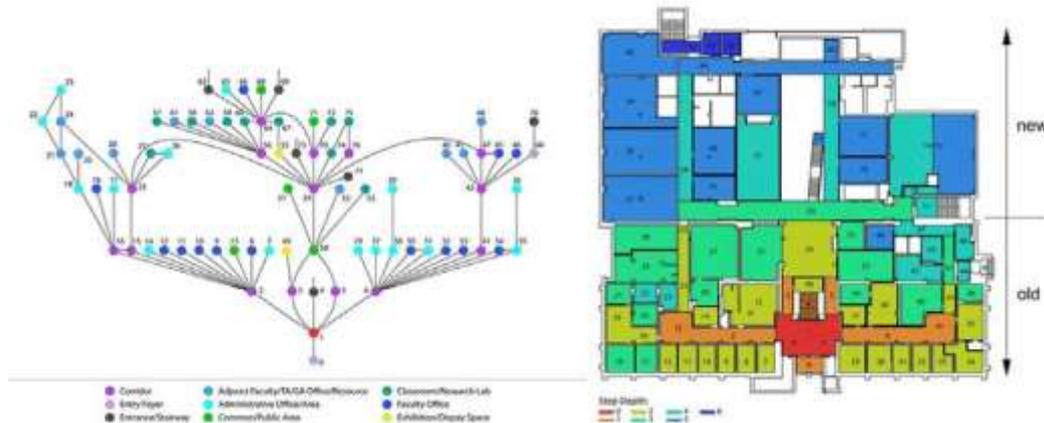


Figure 2. Example of Convex Map (left) and Justified Graph (right) [5]

3. Our Approach

We assume that the architect is putting his philosophy in the building plan. The plan can be an optimized result between the artistic style of the architect and the requirements of the builder. Unlike quantitatively descriptive conditions such as construction cost and land area, the artistic style of the architect is difficult to be expressed as a feature vector of an expression often used in the field of image pattern recognition. Therefore, we would like to describe the styles by way of machine learning of specific architect's drawings themselves. CNN (Convolutional Neural Net) based object detection method, which is a deep learning method for recent image data, has been actively researched and applied to various fields.

We pay attention to detection because we thought this was the beginning of a new challenge to architectural style imitation. We believe that the architect's drawings are the product of the architect's philosophy, the style of his own, the demands of the client, and the reflection of the environment (Figure 1). This is because the drawings made by the architects who are strong enough to be able to anticipate the style of the person who looks at the drawings are different from those of other architects. Analysis of architectural style through drawings is required for mimicry or conversion of architectural style. It is the style analysis of how the wall between the spaces is drawn, how the space between the living room and the kitchen is, how to arrange the size and position of the window.

We are making the drawing style analysis automated. The drawing is accepted as an image input, and the output is discharged in the form that the style of the architect is melted into the system. In this study, in order to automatically recognize the space on the drawing, the space where the furniture exists from the furniture to be placed on the drawing is recognized. For example, when a toilet in a bathroom is detected, the space is classified as a bathroom, and the size, position, and orientation of the bathroom are derived. When a sofa is detected, it is identified as a part of the living room, and when a sink is detected, the space is recognized as a kitchen. Also, the size and location of the windows, which occupy an important position among the architectural elements, can be

detected by a similar process. Based on the results of this detection, it is possible to estimate the architectural style of the drawings based on the theoretical analysis of the architect. The study of how architectural elements such as kitchens, living rooms, and windows are reflected in the architectural style is not included in this paper, but remains to be studied in the future. This paper focuses on the automation of the detection of architectural elements within the drawings, which may be the first step in the execution of a huge project called architectural style analysis.

Our research does not focus on improving the method of detecting objects on images themselves but to utilize learning of building style through detection of building elements in drawings. As a tool for image detection, we utilize object detection API provided by Google.

The TensorFlow Object Detection API is an open source framework built on top of TensorFlow that makes it easy to construct, train and deploy object detection models. Tensorflow Object Detection API is developed to recognize images using Tensorflow. It is provided as a library and provides five models with different accuracy and speed. Even though there is not much concept about machine learning or tensor flow, it can be easily used in library form, and it is easy to create Object Detection System for my scenario by learning by uploading user data directly.

The API has weight values for various deep learning models and builds a detection system suitable for our problem by performing a new learning process based on the existing weights for the new image classifications as initial values. Configuration of architectural elements such as windows, doors, toilets and living room furniture contains the architect's own style. In the future, we will try to develop into a learning algorithm for extracting geometric information between these building elements.

4. Experimental Results

Using the object detection model provided by Google's TensorFlow [6], the image segments related to bathroom furniture in the design drawing are detected by training bathroom furniture image data such as toilet in the building plan. We used this model to test whether it actually works well before learning. For the experiment, OpenCV, PIL, protobuf, lxml, Matplotlib, TensorFlow, tf Slif and Jupyter notebook were used for the experiment.

'ssd_mobilenet_v1_coco' provided by TensorFlow is used as a pre-trained model and this model is trained by MSCOCO Dataset. The ssd_mobilenet_v1_coco model contains a total of 90 classes. Among them, person and kite were detected to show accuracy. It can be confirmed that the object is detected with high accuracy.

TensorFlow's Object Detector API uses TFRecord file format. Therefore, the data to be learned should be modified accordingly. Train data and evaluation data are needed, train data is needed to train the model, and evaluation data is needed to evaluate the accuracy of the trained model.

The first thing we need to do is create our own dataset. Because the Tensorflow Object Detection API uses the TFRecord file format, which is its own file format, we need to convert our dataset to this file format. There are several ways to create a TFRecord file. If you are using Pascal VOC datasets or Oxford Pet datasets, you can use already known Python scripts, but if you use regular image files like ours, then you need to convert them to TFRecord files.

In order to create an input file for the API, we must do the conversion. First, after encoding the RGB image into a jpeg or png file, you need to create a bounding box (consisting of the bottom left point and top right point) for the image and specify the class for the object in the bounding box. We extracted 20 furniture images in bathroom and tagged the data with 'bath'. Since there is no restriction on the direction or size, it is possible to increase the flexibility of the system by enlarging the size of data through

rotation and creating 90 input data. When we expand our research into the kitchen or living room furniture, we similarly collect image data for that class, create a bounding box, and tag the class.

Labeling of the data is done manually using specialized tools such as LabelImg. LabelImg is a graphical annotation tool written in Python that provides a graphical interface using Qt and stores the results in an XML file format. As the last step of data generation, we convert the generated XML file to csv file format and convert it to TFRecords, the objection detection API in Tensorflow input format.

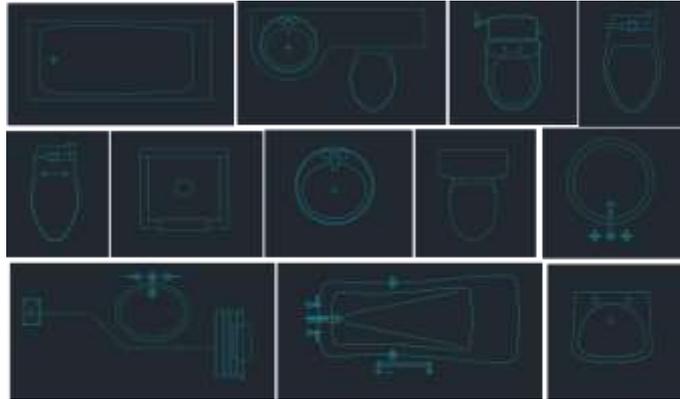


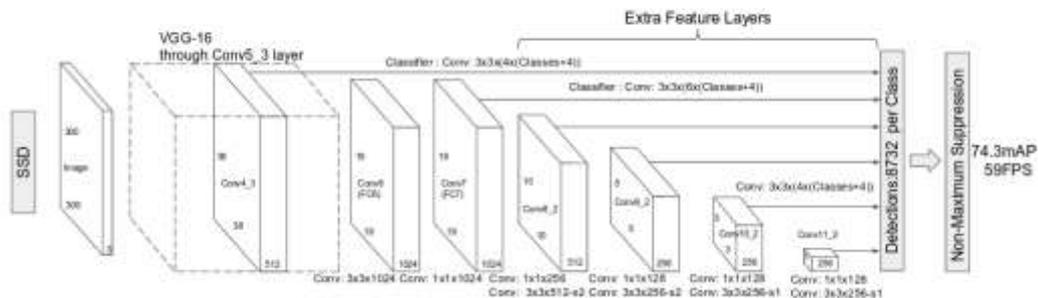
Figure 3. Sample Images of Training Data Set. Total Number of Training Data is 90 and 12 Images Used for Evaluation for Checking the Trained Neural Net Model

After creating the input file required for the Object Detection API, we train our own model. The API provides already-trained examples of object detection algorithms in various ways. The model we used is SSD (Single Shot Multibox Detector) [7], which is successfully applied to various fields. Since our class is one, we set it accordingly (Figure TT). As learning on various neural networks models results in different quality of learning results, starting is used as the default setting of a given model and improves the result quality by adjusting various setting parameters such as learning rate.

The dataset (TFRecord files) and corresponding label map are set as follows:

```

item {
  id: 1
  name: 'bath'
}
    
```



Training is available in local or cloud (AWS, Google Cloud and so on). Tensorflow's proposal suggests that if you have a GPU of 2GB or more, you can do it locally. For Google cloud, you need a YAML configuration file.

The training step where our bathroom data is applied to the pretrained. object detection library and an adequate neural net model is generated for our problem [8] We used `ssd_mobilenet_v1_coco.config` from TensorFlow as pre-trained model. A Relu function is used as the activation function and the standard deviation was set to 0.03 [9]. The loss function uses the sigmoid function for classification loss and the smooth L1 function for localization loss. A batch size was used as 24 for the learning and only 100 steps were repeated.

```
INFO:tensorflow:global step 81: loss = 1.6533 (2.585 sec/step)
INFO:tensorflow:global step 82: loss = 1.6055 (2.699 sec/step)
INFO:tensorflow:global step 83: loss = 1.8596 (2.557 sec/step)
INFO:tensorflow:global step 84: loss = 1.6553 (2.515 sec/step)
INFO:tensorflow:global_step/sec: 0.375034
INFO:tensorflow:global step 85: loss = 1.5570 (3.081 sec/step)
INFO:tensorflow:Recording summary at step 85.
INFO:tensorflow:global step 86: loss = 1.7043 (2.655 sec/step)
INFO:tensorflow:global step 87: loss = 2.4546 (2.572 sec/step)
INFO:tensorflow:global step 88: loss = 1.4774 (2.649 sec/step)
INFO:tensorflow:global step 89: loss = 1.6397 (2.606 sec/step)
INFO:tensorflow:global step 90: loss = 1.8979 (2.526 sec/step)
INFO:tensorflow:global step 91: loss = 1.7746 (2.580 sec/step)
INFO:tensorflow:global step 92: loss = 1.9286 (2.509 sec/step)
INFO:tensorflow:global step 93: loss = 1.7481 (2.553 sec/step)
INFO:tensorflow:global step 94: loss = 1.5094 (2.605 sec/step)
INFO:tensorflow:global step 95: loss = 1.7905 (2.546 sec/step)
INFO:tensorflow:global step 96: loss = 2.2090 (2.609 sec/step)
INFO:tensorflow:global step 97: loss = 2.0067 (2.521 sec/step)
INFO:tensorflow:global step 98: loss = 1.8445 (2.570 sec/step)
INFO:tensorflow:global step 99: loss = 2.0350 (2.491 sec/step)
INFO:tensorflow:global step 100: loss = 1.9052 (2.610 sec/step)
INFO:tensorflow:Stopping Training.
INFO:tensorflow:Finished training! Saving model to disk.
```

Figure 5. The Learning Process stops in 100 Steps with Total Loss 1.9052. Above stats Show the Optimization for Learning Finishes with Some Oscillation

And the hit value came out to about 0.653 for evaluation step. It is confirmed that object detection can be performed with a small number of train data and steps using the pre-trained model.

In the experiment, it is found that objects can be detected in the building plan even with small train data. In addition to the bath, you will be able to train various cases such as dining room or kitchen. And by doing such detection, it will be possible to extract the style of the architect by checking the layout of the furniture, windows, doors, *etc.* in the architect's design drawings, and the machine will be able to imitate the style. However, it takes a lot of resources and computational time because it requires a lot of computations of learning. Deep learning using Google Cloud ML [10] would be a good solution for such situations of humble computational resources.

5. Conclusion

In this paper, we have challenged a new problem of architect style learning and conducted a study to detect architectural elements in the first stage of the drawing. The idea of learning the architect's creative style from the drawing means that the paradigm of artificial intelligence enters into the area of creation which is regarded as the unique area of the artist such as painting image or music score. We hope that this challenge will not end in architecture, but will extend to creative intentions and styles in product design and other three-dimensional design areas.

We will extend the scope of detection to other households, based on the achievements of this study in bathroom furniture. As mentioned at the beginning, recognition of space in drawings is an essential element in the analysis of drawings. In addition, we will analyze the works that architects have done in the past and analyze the style of specific architects based on the interpretations of each architect in the field of architectural analysis. In the process, we expect to be able to describe the architect's style through quantitative data of architectural elements such as bathrooms, living rooms, bedrooms, and windows. Given the drawings, it is the next problem that makes it easy to sort out who's work. In addition, we expect to be able to provide a system that outputs the drawings designed by a specific architect who wants to be given only volume ratio, volume, *etc.* based on the study of converting a given picture or image into the style of a specific painter.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) Funded by Ministry of Education (2017R1D1A3B03034586)

This paper is a revised and extended version of a paper entitled [Aesthetic Intension Extraction in Architecture based on Deep Learning Framework] presented at [The 9th International Mega-Conference on Future Generation Information Technology (FGIT 2017), Daejeon University, Korea, Dec.21~23, 2017].”

References

- [1] S. Ren, “Faster R-CNN: Towards real-time object detection with region proposal networks”, *Advances in neural information processing systems*, (2015).
- [2] J. Redmon, “You only look once: Unified, real-time object detection”, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2016).
- [3] L.A. Gatys, “Alexander S. Ecker, and Matthias Bethge: Image style transfer using convolutional neural networks”, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2016).
- [4] W.-H. Lee, “Aesthetic Intension Extraction in Architecture based on Deep Learning Framework”, *Advanced Science and Technology Letters*, Vol.148, (FGIT), (2017), pp.11-14
- [5] Y. McLane, “Choreographing collaborative academic experiences: The 'quiet building' and the 'airport lounge’”, *Proceedings of the 10th International Space Syntax Symposium*, (2015).
- [6] <https://www.tensorflow.org/>.
- [7] W. Liu, “Ssd: Single shot multibox detector, European conference on computer vision”, Springer, Cham, (2016).
- [8] N. Jaitly, “Application of pretrained deep neural networks to large vocabulary”, *speech recognition, Thirteenth Annual Conference of the International Speech Communication Association*, (2012).
- [9] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition”, *arXiv preprint arXiv:1409.1556*, (2014).
- [10] <https://cloud.google.com/ml-engine/>.

Authors



Jinho Park, he is an associate Professor at Soongsil University, Korea, South Korea



Doyoung Yoon, he is an undergraduate Student at Soongsil University, Korea, South Korea



Woo-Hyoung Lee, he is an associate Professor at Namseoul University, Cheonan South Korea