HIERARCHICAL (MULTI-LABEL) ARCHITECTURAL IMAGE RECOGNITION AND CLASSIFICATION

JIELIN CHEN[1], RUDI STOUFFS[2] and FILIP BILJECKI[3]
[1,2,3]*Department of Architecture, National University of Singapore*
[1]*chen.jielin@u.nus.edu* [2,3]*{stouffs|filip}@nus.edu.sg*

Abstract. The task of architectural image recognition for both architectural functionality and style remains an open challenge. In addition, the paucity of well-organized, large-scale architectural image datasets with specific consideration for the domain of architectural design research has hindered the exploration of these challenging tasks. Drawing upon images from the professional architectural website Archdaily®, and leveraging state-of-the-art deep-learning-based classification models, we explore a hierarchical multi-label classification model as a potential baseline for the task of architectural image classification. The resulting model showcases the potential for innovative architectural discipline-related analyses and demonstrates some heuristic insights for visual feature extraction pertaining to both architectural functionality and architectural style.

Keywords. Image recognition; hierarchical classification; multi-label classification; architectural functionality; style.

1. Introduction

The emerging awareness, and pertinent discussion, regarding the application of machine learning techniques is permeating the architectural discipline of both its methodology and epistemology. Architectural image classification models with high performance show potential as informative tools for a series of architecture-related tasks. For example, classifying the functionality of individual buildings can be useful for urban utility planning and population density mapping at a finer level of urban intrinsic scale (Kang et al. 2018; Hoffmann et al. 2019); identifying visual features of building instances can assist the investigation of the impact of built environment characteristics (Nguyen et al. 2018; von Platten et al. 2020); and recognition of emerging architectural styles can provide novel insights into the trend of modern architectural design practice. Nevertheless, architectural image classification can deviate from conventional image recognition tasks due to the convoluted inter-class relationships between different architectural categories and styles (Xu et al. 2014), as there is no standard criterion regarding the definition of architectural types and styles concerning visual features, and some architectural types and styles can be interdependent and the corresponding latent features may not be identically distributed. Hitherto, only limited efforts have been made to address the task of individual building instance classification

PROJECTIONS, Proceedings of the 26th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA) 2021, Volume 1, 161-170.

(Kang et al. 2018). Meanwhile, although existing architectural style-focused datasets can be adapted to some interesting machine learning-based applications such as style transformation, the predefined styles involved in existing datasets are mostly of historical significance and might have limited application potentials in real-world design scenarios. Although it would be worthwhile to explore the variety of modern architectural styles that are somehow ill-defined in current architectural literature, few previous studies have explored the task of architectural style prediction with a perspective of modern architectural design practice.

Thus, the task of architectural image recognition for both architectural functionality and style remains an open challenge. However, the paucity of well-organized, large-scale architectural image datasets has hindered the exploration of these challenging tasks. Even though there are generic data banks available with tagged images, it can still be tricky to find specific datasets of architectural images for various purposes and the quality of images is not guaranteed with respect to architectural design. Hence, there is a necessity for new large-scale architectural image datasets with hierarchical labelling and different levels and details of annotations, which could be useful for the training of deep neural networks or other machine learning techniques for architectural design research. Shalunts et al. (2011) have collected a small dataset with 400 building facade images labelled by architectural styles for the classification task of cultural heritage buildings. Llamas et al. (2017) have compiled a publicly available dataset with more than 10,000 images sorted in 10 types for classifying architectural elements of interest in imagery of heritage buildings. Xu et al. (2014) have extracted and fine-tuned an architectural style dataset from Wikimedia with 25 architectural style classes tailored for architectural style classification, and each class has images ranging from 60 to 300 with a total number of roughly 5,000. Recently, Kang et al. (2018) have built a dataset to facilitate the training and evaluation of building instance classifiers using street view images, while using geographic information retrieved from online map services for labelling. The dataset has a training set of size 17,600 and a test set of 2,058. However, the dataset only possesses 8 classes, and the environmental context of the images retrieved from street view websites is somehow homogeneous, which has made the image dataset limited pertaining to the level of diversity.

We compiled an architectural image database called AIDA, short for Annotated Image Database of Architecture, composed of building imagery with high-diversity and high-coverage for general-purpose deep learning-based model training. The new dataset provides an enhanced platform for the evaluation of the performance of existing deep learning-based models, as well as encouraging the creation of new ones. We also offer a series of multi-label architectural image classifiers with integrated classification labels, including scene classes (indoor and outdoor-street-level) and architectural functionality categories. The obtained architectural image classifier showcases the potential for many innovative architectural discipline-related analyses. Also, it provides some heuristic insights with respect to visual feature extraction in the context of architectural design research.

2. Construction of AIDA

To ensure the quality of imagery concerning architectural design and satisfy the requirement of a broad spectrum of coverage, images are retrieved from professional architectural website Archdaily®, an architecture projects broadcasting website with probably the largest online repository of architecture projects worldwide. The crawled images, one image per architectural project, have been manually filtered to meet specific requirements for training tasks for architectural design research: the photos need to be real-world photography and need to be focused on the architecture. Unqualified or irrelevant images have been discarded. Images retrieved from Archdaily® have been annotated with ground truth category labels acquired from the website. To ameliorate class imbalance, architectural categories with too few or too many images have been omitted to compose a condensed image database. Images are further categorized into two scene classes: outdoor-street-level and indoor, with 25 architectural categories each (image samples are shown in Figure 1). The number of images in each architectural category of each scene class varies from 20 to 1,400, and the total number of images in AIDA is 14,659 (Figure 2). Noticeably, the underlying inter-relationships between different architectural categories might distinguish the newly collected database from conventional scene classification datasets.

Figure 1. Image samples from two scene classes, outdoor-street-view and indoor, and various architectural categories of AIDA (source: https://www.archdaily.com/).

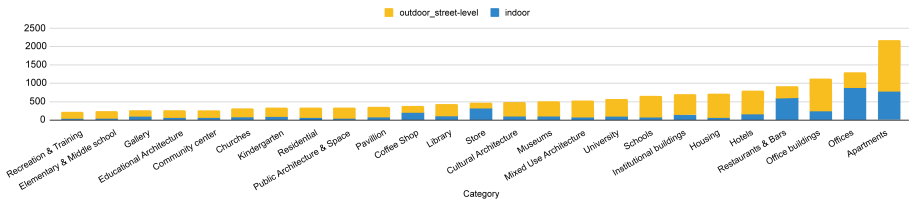


Figure 2. Number of images per category in AIDA, sorted in ascending order; AIDA contains 14,659 images from 25 architecture categories.

3. AIDA-CNNs: hierarchical (multi-label) architectural image classification

Compared with flat classification tasks, hierarchical classification can be a more efficient approach to organize the enormous amount of information involved and can be cast to more real-world applications; classes are pre-established as different levels of hierarchy, either a tree or a directed acyclic graph (DAG) structured class taxonomy (Silla & Freitas 2011). The AIDA database has a DAG class taxonomy as each child class—architectural category—can be directed back to more than one parent class—scene class. Hence, the hierarchical classifier might be more suitable for the AIDA database compared with flat classifiers.

Most approaches in the context of hierarchical classification can be regarded as multi-label classification and categorized into local or global classifiers. Local classifiers explore the class structure in a top-down manner with a series of classifiers; global classifiers employ a single classifier dealing with the entire class structure (Silla & Freitas 2011). The local classifier per parent node (top-down) approach adopts one multi-label classifier for each parent class in the hierarchy to distinguish between its child classes or, alternatively, a multi-label classifier for each hierarchical level. Instead, the global classifier approach takes into account the dependencies between classes in a more straightforward way and a single yet relatively complex classification model is constructed, treating the class hierarchy as a whole for a single run of the classification algorithm (Freitas & Carvalho 2007). Compared with the modularity for local training of the classifier, global classifiers have the advantage of learning a global model for all the classes in a single run yet adding complexity to the adopted model.

To fuse the scene classification and architectural category classification as an integrated task, we adopt a global classifier in the context of hierarchical classification as the basic, multi-label classification framework. Figure 3 illustrates the proposed hierarchical image classification model framework: an integrated CNN model as the global classifier takes in the labeled architectural images and produces predicted labels of both the scene classes and the architectural categories, which are then separately interpreted as two hierarchical classifiers.

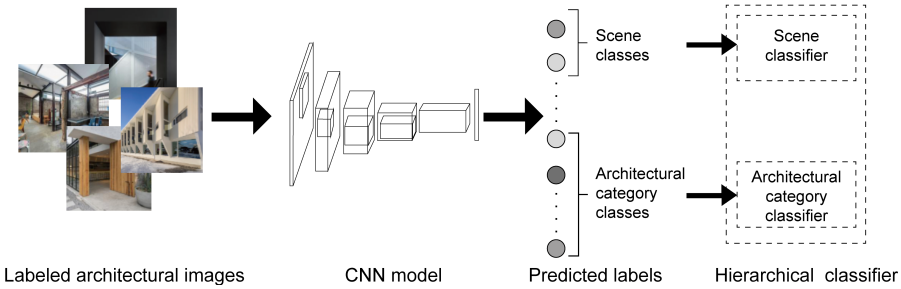


Figure 3. The proposed hierarchical image classification model framework.

3.1. TRAINING

We chose two state-of-the-art CNN architectures for image classification tasks, namely ResNeXt (Xie et al. 2017) and DenseNet (Huang et al. 2017), to construct the baseline CNN models. While adopting the basic model structure of ResNeXt and DenseNet, the output layers of both models are modified using a sigmoid function for the output layer to suit the multi-label classification task. Such modification also enables the analysis of inter-relationships of architectural classes probabilistically and offers a latently rational explanation of the gradual transition and mixture of visual architectural features with soft probabilistic assignments.

The image database is still not sufficiently large to properly train the selected models with over millions of parameters from scratch; at least an order of magnitude more instances relative to the trainable parameters of the model would be sufficient (Google Developers, n.d.). Therefore, the convolutional layers of the networks have been trained using the transfer learning approach. Fine-tuning a pretrained CNN for new training tasks with novel datasets has been proven to be efficient, as local features like edges and corners generated by the bottom layers of the neural network are usually similar for different types of imagery. In contrast, the high-level features extracted by the top layers are task-dependent. For the model training, 11,730 images from the dataset have been randomly selected for training and 2,929 for testing, while 20% of images from the training samples have been selected as validation data to monitor the training status of the networks. For the experimental implementation, we chose the corresponding network version of the ResNeXt and DenseNet models with relatively better performance based on the evaluation provided by PyTorch (Paszke et al. 2019), respectively, ResNeXt-101 and DenseNet-161, where the numbers 101 and 161 in the nomenclature denote the depth of the specific version of the network models.

Both networks are initialized with corresponding model checkpoints provided by PyTorch, which were pre-trained on the ImageNet database (Deng et al. 2009), and the output layers were initialized in a random manner by adopting a uniform distribution. The training adopted a batch size of 32 and used the adaptive moment estimation algorithm (Kingma & Ba 2014) with a learning rate of $\alpha = 10^{-3}$, exponential decay rate for the first moment estimates $\beta_1 = 0.9$, exponential decay rate for the second-moment estimates $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$ for training. The binary cross-entropy loss function was used and a drop out rate of 20% was adopted for the neurons of fully connected layers. Data augmentation was used for the training data with the following settings: (a) random crop size of 224×224 pixels with resize range of 0.8-1.0 of initial input size 256×256 pixels; (b) random rotation with a range from -15 to 15 degrees; (c) random horizontal flip with a probability of 50%. All training and testing were performed with PyTorch (Paszke et al. 2019) on 2 Nvidia Tesla V100 32GB GPUs.

Three classification accuracy metrics have been calculated using scikit-learn (Pedregosa et al. 2011), including weighted precision, weighted recall and weighted F1 score, which are typically used metrics for multi-label targets. The nomenclature "weighted" indicates that the averaging performed on the calculated metric is weighted by the number of true instances of each label, which accounts

for the impact of latent label imbalance. Precision p = *true positive* / (*true positive* + *false positive*) is the ratio of true positive predictions to the total predicted positive instances. Recall r = *true positive* / (*true positive* + *false negative*) is the ratio of true positive predictions to all instances in the corresponding class. F1 score $F_1 = 2(rp)/(r+p)$ is the weighted average of precision and recall, and considers both false positive and false negative instances, which can be useful if the class distribution is uneven. As can be seen from the calculated accuracy metrics in Figure 4, both networks have fluctuations for the weighted F1 score at the early stage, plausibly caused by the uneven distribution among different classes. Meanwhile, DenseNet-161 has outperformed ResNeXt-101 based on the three accuracy metrics (weighted precision, weighted recall and weighted F1 score) calculated over different training epochs.

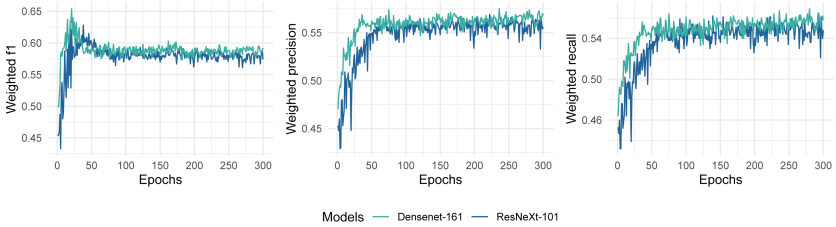


Figure 4. DenseNet-161 obtains slightly higher accuracy compared to ResNeXt-101.

3.2. TESTING

To testify the model performance on the two hierarchical levels, we further evaluate the trained classifiers on the scene classes and architectural categories separately using the test set. Figure 5 illustrates the corresponding overall accuracy and accuracy of each scene class at different training epochs of ResNeXt-101 and DenseNet-161. DenseNet-161 achieves slightly better performance on the scene classification task with 96% accuracy compared to ResNeXt-101 with 95% accuracy.

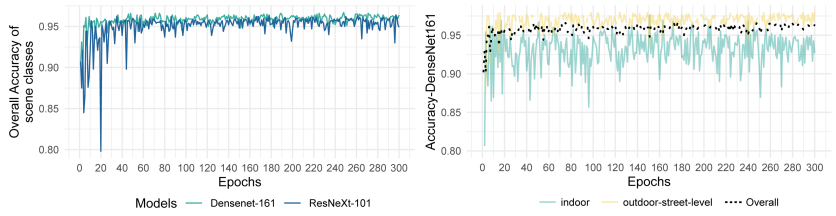


Figure 5. Comparison of the overall accuracy of the scene classes between ResNeXt-101 and DenseNet-161(left) and accuracy of each scene class of DenseNet-161 (right).

A comparison of the overall accuracy of the architectural categories between ResNeXt-101 and DenseNet-161 (Figure 6 left) shows that DenseNet-161 also slightly outperforms ResNeXt-101 on the architectural image classification task.

It is worth noting that the overall accuracy of architectural category classification fluctuates at early epochs, plausibly caused by the divergent accuracy value among different architectural categories during initial epochs (Figure 6 right). The discrepancy of accuracy between different architectural categories gradually decreases as the number of epochs increases, which might also contribute to the fluctuation phenomenon of the weighted F1 score over all target classes at the early stage.

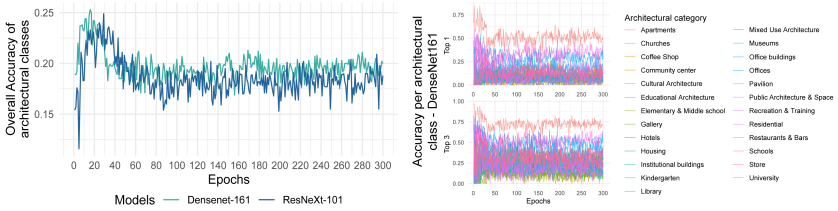


Figure 6. Comparison of the overall accuracy of architectural categories between ResNeXt-101 and DenseNet-161 (left) and top-1, top-3 accuracy per category of DenseNet-161 (right).

To further examine this phenomenon, the mean and standard deviation of the accuracy per architectural category are calculated for both ResNeXt-101 and DenseNet-161 (Figure 7). The mean accuracy per architectural category increases while the standard deviation decreases with the increase of the number of epochs as anticipated. DenseNet-161 has a higher mean and lower standard deviation of accuracy per architectural category during the early training epochs, while the difference gradually eliminates as the number of epochs increases.

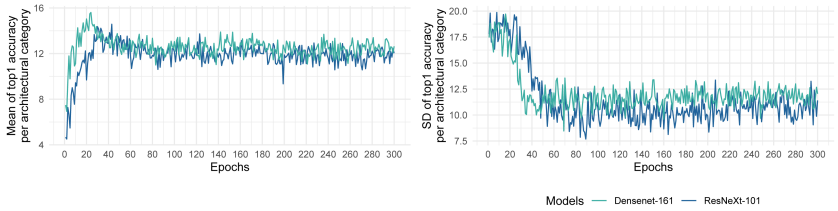
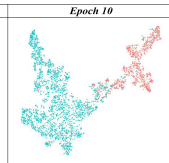
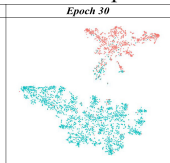
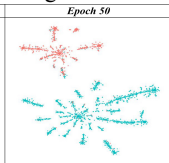
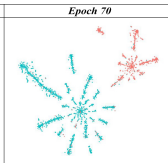
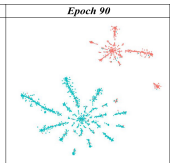
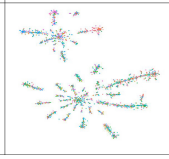
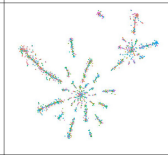
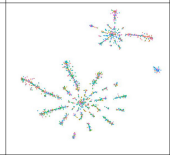


Figure 7. Comparison of the mean (left) and standard deviation (right) of top-1 accuracy per architectural category between ResNeXt-101 and DenseNet-161.

3.3. EVALUATION

Table 1 uses t-SNE (Van der Maaten & Hinton 2008) to visualize the high-dimensional space of the image classification manifolds by giving each predicted datapoint a location in a two-dimensional map. We demonstrate the predicted class projection via t-SNE with DenseNet-161 trained at different epochs using the full dataset and the test set respectively: the separation between different classes becomes gradually more pronounced as the number of epochs increases. The separation between the two scene classes is already distinguishable after 30 epochs, while for the architectural categories, the situation is more convoluted.

Table 1. Predicted class projection via t-SNE with DenseNet161-based hierarchical multi-label classifier trained at different epochs using the full dataset and the test set of AIDA.

	Epoch 10	*Epoch 30*	*Epoch 50*	*Epoch 70*	*Epoch 90*
Scene class					
Architectural class					

To examine the disentanglement between architectural categories, the normalized confusion matrix is plotted based on the AIDA test set evaluated with DenseNet-161 trained at epoch 70 (Figure 8). Some pairs of architectural classes have entangled relationships with each other, indicating the complex inter-class relationships between different architectural categories, as mentioned in the introduction.

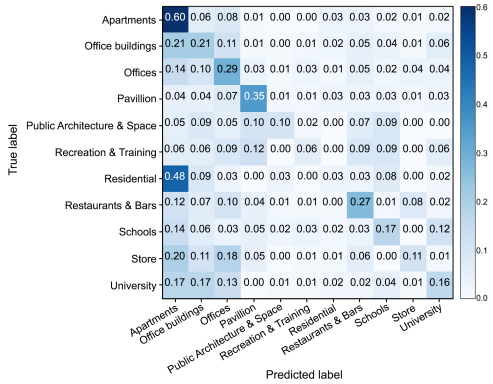


Figure 8. Normalized confusion matrix for architectural category prediction with DenseNet-161 trained until epoch 70 (showing only 11 selected categories), demonstrating the convoluted inter-class relationships between different architectural categories.

To further examine the latent stylistic relationship between architectural categories, we projected the prediction of each image of the "Apartments" category in the test dataset with DenseNet-161 at epoch 70 using t-SNE and examined a series of images from different prediction clusters. Surprisingly, the projected prediction mapping reveals some latent inter-class style relationships. We used the Gradient-weighted Class Activation Mapping (Grad-CAM) proposed by Selvaraju et al. (2017) to produce coarse localization maps with highlights of important discriminative regions of the image which correspond to the predictive decision of interest. Red regions of the heatmap correspond to high scores for class prediction significance (Figure 9).

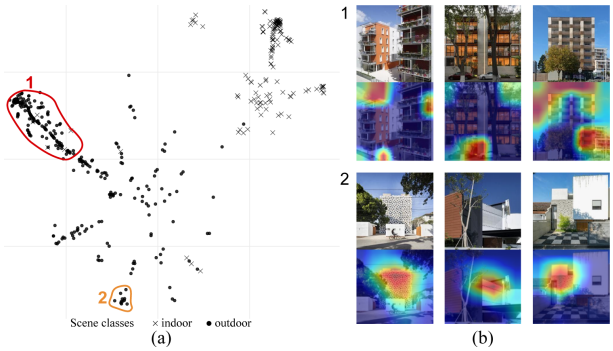


Figure 9. (a) Prediction of "Apartments" category with DenseNet-161 at epoch 70 projected using t-SNE; (b) Sample images with feature activation maps produced using Grad-CAM: images from clusters 1 have been correctly classified as "Apartments", while images from clusters 2 have been incorrectly classified as "Offices".

We note that the architectural category classification has an intrinsic relationship with some stylistic characteristics of the corresponding architectural image: the images mistakenly classified as another architectural category in clusters 2 are in possession of some common visual features, such as the solid white cubic geometry recurring in cluster 2. Meanwhile, the images which are correctly classified are also in possession of some similar visual features, such as the grid pattern of images from cluster 1.

4. Discussion and summary

We explored two multi-label architectural image classifiers with integrated classification labels, including scene classes and architectural functionality categories, trained on a new architectural image dataset with hierarchical labelling. The resulting model showcases the potential for innovative architectural discipline-related analyses.

The latent stylistic relationship between different architectural categories has revealed heuristic insights with respect to visual feature extraction in the context of architectural design research. The architectural image classification models can capture some deeper representations of higher-level visual features which can be related to the interpretation of architectural stylistic characteristics. Such property can be leveraged for architectural style identification and induction. The classification models can also be leveraged to develop an architectural style relationship network and provide architectural style analysis for individual buildings, which might distinguish itself from existing architectural style classification models relying on predefined historical styles with limited application potentials in real-world design scenarios.

5. Data availability statement

Some or all data and code that support the findings of this project are available from https://dataverse.harvard.edu/dataverse/AIDA_AIDA-CNNs.

References

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K. and Li, F.F.: 2009, Imagenet: A large-scale hierarchical image database, *2009 IEEE conference on computer vision and pattern recognition*, 248-255.

Google Developers, initials missing: n.d., "The Size and Quality of a Data Set" . Available from <https://developers.google.com/machine-learning/data-prep/construct/collect/data-size-quality> (accessed 1st November 2020).

Freitas, A. and Carvalho, A. 2007, A tutorial on hierarchical classification with applications in bioinformatics, *in* D. Taniar (ed.), *Research and trends in data mining technologies and applications*, IGI Global, 175-208.

Hoffmann, E.J., Wang, Y.Y., Werner, M., Kang, J. and Zhu, X.X.: 2019, Model fusion for building type classification from aerial and street view images, *Remote Sensing*, **11**(11), 1259.

Huang, G., Liu, Z., Van Der Maaten, L. and Weinberger, K.Q.: 2017, Densely connected convolutional networks, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4700-4708.

Kang, J., Körner, M., Wang, Y., Taubenböck, H. and Zhu, X.X.: 2018, Building instance classification using street view images, *ISPRS journal of photogrammetry and remote sensing*, **145**, 44-59.

Kingma, D.P. and Ba, J.: 2014, Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980*.

Llamas, J., M Lerones, P., Medina, R., Zalama, E. and Gómez-García-Bermejo, J.: 2017, Classification of architectural heritage images using deep learning techniques, *Applied Sciences*, **7**(10), 992.

Van der Maaten, L. and Hinton, G.: 2008, Visualizing data using t-SNE, *Journal of machine learning research*, **9**(Nov), 2579-2605.

Nguyen, Q.C., Sajjadi, M., McCullough, M., Pham, M., Nguyen, T.T., Yu, W.J., Meng, H.W., Wen, M., Li, F.F. and Smith, K.R.: 2018, Neighbourhood looking glass: 360º automated characterisation of the built environment for neighbourhood effects research, *J Epidemiol Community Health*, **72**(3), 260-266.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N. and Antiga, L.: 2019, Pytorch: An imperative style, high-performance deep learning library, *Advances in neural information processing systems*, 8026-8037.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R. and Dubourg, V.: 2011, Scikit-learn: Machine learning in Python, *the Journal of machine Learning research*, **12**, 2825-2830.

von Platten, J., Sandels, C., Jörgensson, K., Karlsson, V., Mangold, M. and Mjörnell, K.: 2020, Using Machine Learning to Enrich Building Databases—Methods for Tailored Energy Retrofits, *Energies*, **13**(10), 2574.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D. and Batra, D.: 2017, Grad-cam: Visual explanations from deep networks via gradient-based localization, *Proceedings of the IEEE international conference on computer vision*, 618-626.

Shalunts, G., Haxhimusa, Y. and Sablatnig, R.: 2011, Architectural style classification of building facade windows, *International Symposium on Visual Computing*, 280-289.

Silla, C.N. and Freitas, A.A.: 2011, A survey of hierarchical classification across different application domains, *Data Mining and Knowledge Discovery*, **22**(1-2), 31-72.

Xie, S., Girshick, R., Dollár, P., Tu, Z. and He, K.: 2017, Aggregated residual transformations for deep neural networks, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1492-1500.

Xu, Z., Tao, D., Zhang, Y., Wu, J. and Tsoi, A.C.: 2014, Architectural style classification using multinomial latent logistic regression, *European Conference on Computer Vision*, 600-615.