Recognizing Buildings through Deep Learning: A Case Study on Half-timbered Framed Buildings in Calw City

Konstantinos Makantasis[1], Nikolaos Doulamis[2] and Athanasios Voulodimos[2]
[1]*Technical University of Crete, Chania, Greece*
[2]*National Technical University of Athens, Athens, Greece*

Keywords: Deep Learning, Building Recognition, Urban Cultural Heritage.

Abstract: Automatic detection and recognition of specific types of urban buildings is extremely important for a variety of applications ranging from outdoor urban reconstruction to navigation. In this paper we propose a system for the automatic detection and recognition of urban buildings. Most of the existing work relies on the exploitation of handcrafted features for recognizing buildings. However, due to their complex structure it is rarely a priori known which features are important for the recognition task. Our method overcomes this drawback by exploiting a deep learning framework, based on convolutional neural networks, which automatically construct highly descriptive features directly from raw data. We evaluate the performance of our method on the recognition of half-timbered framed buildings in Calw city in Germany.

1 INTRODUCTION

Urban buildings consist an integral part of cultural heritage. They shape the sense of belonging somewhere, of social traditions, of cultural identity of a history spanning centuries. Urban buildings are material witnesses, which can be questioned and analyzed over and over again. Therefore, automatic detection and recognition of specific types of urban buildings is extremely important for disseminating cultural heritage to general audience (Bres and Tellez, 2006; Olojede and Suleman, 2015), for outdoor urban reconstruction (Xiao et al., 2009; Müller et al., 2007), navigation (Schindler et al., 2008; Baatz et al., 2010) and scene understanding (Singhal et al., 2003) applications.

Automatic detection and recognition of specific urban building types can be seen as a classification problem. The detection task requires the classification of pixels into two different classes, i.e. class of pixels that depict the object of interest and class of pixels that depict something irrelevant.

Most of the existing works on automatic detection and recognition of building types adopt the conventional pattern recognition paradigm, which consists of two separate steps; firstly, complex handcrafted features need to be constructed, in order to accurately describe the buildings, and, secondly, these features are used to learn classifiers, which conduct the detection and/or recognition task.

The most common used features for describing objects of interest, in our case buildings, rely on the exploitation of local features, i.e. features that describe the local properties of objects of interest. Having constructed local features that describe the objects of interest, the detection and/or recognition of similar objects in unseen images is conducted by detecting similar features the same features in these images.

In the works of (Ali et al., 2007a; Ali et al., 2007b) multi-scale Haar wavelet features are used to learn a cascade classifier, based on AdaBoost method, for automatically recognizing building windows. The authors of (Shechtman and Irani, 2007) propose a method that exploits the local self-similarities to match complex visual data. In particular, their method correlates a patch centered at the point of interest with a larger surrounding region and use the maximal correlation values within log-polar bins as descriptor, emphasizing this way the local shape properties of objects of interest. In the works of (Tell and Carlsson, 2000; Tell and Carlsson, 2002) a pixel intensity based descriptor is utilized to described the local properties of objects of interest and then a voting scheme is exploited to match the descriptors between different images. The work of (Wendel et al., 2010) extends the works of (Tell and Carlsson, 2000; Tell and Carlsson, 2002) by extracting color intensity profiles for points of interest that are lying on straight lines. These pro-

Makantasis K., Doulamis N. and Voulodimos A.
Recognizing Buildings through Deep Learning: A Case Study on Half-timbered Framed Buildings in Calw City.
DOI: 10.5220/0006347204440450

Figure 1: Half-timbered framed buildings in Calw.

files are used as descriptors that are being matched using a kd-tree algorithm (Friedman et al., 1977).

Another common approach for describing the local properties of objects of interest rely on the exploitation of Scale Invariant Feature Transform (SIFT) (Lowe, 2004). However, SIFT features and all other methods that are based on the utilization of local descriptors are problematic in urban scenes due to the high degree of symmetry and repetition in these environments. In particular, these factors impede finding a geometrically consistent match between reference and query images, which is crucial for the detection and recognition of specific types of buildings. Furthermore, due to the high diversity of visual content it is rarely a priori known which features are important for the building detection and/or recognition task.

In contrast to the aforementioned approaches, which follow the conventional paradigm of pattern recognition, we adopt a *deep learning* framework for the recognition of specific types of buildings. Deep learning models (Hinton and Salakhutdinov, 2006; Hinton et al., 2006; Bengio et al., 2007) are a class of learning machines that can learn a *hierarchy* of features by building high-level features from low-level ones, thereby automating the process of feature construction for the problem at hand. Techniques based on deep learning have already shown promising results both for the detection of particular objects, like man-made ones (Mnih and Hinton, 2012; Makantasis et al., 2015) or vehicles (Montavon et al., 2012).

In a similar way, we propose a deep learning framework for the recognition of specific types of urban buildings. By following a deep learning approach we are able to bypass the tedious, time consuming and questionable task of constructing handcrafted features for describing building characteristics.

In particular, we propose the exploitation of a Convolutional Neural Network (CNN), which conducts the task of high-level features construction and a Multi-Layer Perceptron (MLP), which is responsible for the recognition task. Under such a formulation, the developed system is able to automatically construct high level features for describing the buildings, while at the same time achieves real-time predictions due to the feedforward nature of CNNs and MLPs. The performance of the proposed approach is demonstrated on the recognition of half-timbered framed buildings in Calw city (see Fig. 1).

The rest of the paper is organized as follows; section 2 presents the system requirements and section 3 presents the overview of the proposed system. Section 4 describes the creation of the dataset for training the proposed deep learning framework, whose architecture is presented in section 5. Section 6 presents the experimental results regarding the performance of the proposed system, while section 7 concludes this work.

2 SYSTEM REQUIREMENTS

Our goal is to develop a vision based system able to recognize specific type of buildings. We demonstrate the performance of the proposed system on the recognition of half-timbered framed buildings in Calw city (see Fig. 1).

Such a system must fulfill the following requirements; (i) it should be able to recognize buildings based exclusively on visual cues. This implies that no textual or GPS data should be employed during the recognition task. (ii) The developed system should be scale and perspective invariant, i.e. it should be able to recognize the specific type of buildings irrespectively of the buildings area on image plane and their angle of view. Finally, (iii) the building recognition task should be successful even on images that have been taken on the wild; i.e. images that are captured by pedestrians and/or tourists and images that are available over web repositories, such as Flickr, Picasa, etc.

In the following we present our approach overview, having in mind the aforementioned system requirements.

(a) (b)

Figure 2: (a) Original and (b) manually annotated image.

3 APPROACH OVERVIEW

Having the aforementioned system requirements in mind, we follow a learning based approach for the recognition of half-timbered framed buildings in Calw city. Every learning machine requires a labeled dataset, whose samples should be described by high discriminative features. This includes two different things; firstly we have to obtain or create an image dataset and secondly, we have to pre-process the images in order to construct features that accurately describe their visual content.

As mentioned before, we adopt a deep learning framework for the recognition of half-timbered framed buildings. Due to the fact that deep learning models can automatically construct high level features form low level ones, we are able to bypass the data pre-processing task for constructing handcrafted features.

Specifically, we exploit Convolutional Neural Networks (CNN). CNNs consist a type of deep models, which apply trainable filters and pooling operations on the raw input, resulting in a hierarchy of increasingly complex features. Although, it has been shown that CNNs can achieve superior performance on visual recognition tasks without relying on handcrafted features, due to their nature, they produce global image features. This implies that CNNs can respond whether or not a half-timbered frame building is present in a picture.

However, the recognition problem includes not only the task for deciding if the objects of interest are present in a scene but also their localization on image plane. Due to this fact, instead of using the whole image as input to the network, we feed the network with image patches. This way, the CNN decides if an image patch depicts the object of interest or some part of it. We use this procedure along with sliding windows to make a decision for every patch in the image. Then, a voting mechanism based on patches classification results is used to classify image pixels.

4 DATASET CREATION

For recognizing objects of interest, in our case half-timbered framed buildings, in an image, we classify image patches. Successful classification of patches is based on the successful training of the deep learning architecture i.e. the successful training of the CNN and MLP. This requires the creation of a dataset for training, which includes patches that depict (positive samples) and not depict (negative samples) the object of interest or some part of it.

For creating the dataset we used one hundred images captured in Calw city. These images were manually annotated i.e. for each image a mask around the object of interest was created (see Fig. 2).

Using the annotated images we croped a predefined number of square patches from each one of the original images. These patches are labeled as positive if their centered pixel is located in the mask, i.e. black pixel in annotated image, and the total number of masked pixels is larger than 66.6% of the total number of pixels in the patch. Similarly, the patches whose centered pixel is located outside of the mask and the total number of unmasked pixels is larger than 66.6% of the total number of pixels in the patch are labeled as negative samples. Examples of positive and negative samples are presented in Fig. 3.

As mentioned before, a predefined number of positive and negative samples (patches) were cropped from each image. In order to made our system scale invariant the size of the patches was randomly vary-

(a) (b)

Figure 3: (a) Negative and (b) positive samples.

ing between 256×256 and 512×512 pixels. Finally, all cropped patches were resized to 32×32 pixels in order to be fed as input to the CNN.

5 DEEP LEARNING ARCHITECTURE

In this work the learning machine that is responsible for constructing features in order to describe patches is a CNN. CNNs consist a type of deep models, which apply trainable filters and pooling operations on the raw input, resulting in a hierarchy of increasingly complex features. On top of the CNN a fully-connected feed-forward neural network (MLP), whose input corresponds to the high-level features constructed by the CNN, was placed to conduct the classification task.

The CNN consists of thee convolutional layers. The first convolutional layer consists of 9 trainable filters of size 7×7, the second consists of 12 trainable filters of size 5×5 and the third consists of 15 trainable filters of size 3×3.

Each convolutional layer is followed by a max-pooling operator of size 2×2. Max pooling is done by applying a max filter to non-overlapping sub-regions of the initial representation, i.e. extracts the maximum element of non-overlapping sub-regions of size 2×2 of the image that resulted by the application of the trainable filters.

Finally, the fully-connected feed-forward network contains one hidden layer with 45 neurons. This architecture results in a network with 6491 trainable parameters. The architecture of the proposed deep learning system is schematically represented in Fig. 4.

5.1 Architecture Training

During the training of the architecture, we want to select a function $f : \Re^{32\times32} \rightarrow \{0,1\}$, i.e. a function f that maps each image patch $x \in \Re^{32\times32}$ to object of interest and non object of interest classes. The form of the function f is the one that corresponds to the aforementioned architecture and it can be fully identified by the set, w of trainable parameters.

Estimating the optimal set w of architecture parameters involves minimizing a loss function. In the case of binary classification, it is very common to use the negative log-likelihood as the loss. This is equivalent to maximizing the likelihood of the training data set under the model parameterized by w.

The minimization task, and thus the parameter estimation, takes place by exploiting the back propagation algorithm and choosing the stochastic gradient descent as the optimization method.

5.2 Early Stopping Criteria

The training phase is terminated either when a maximum number of training epochs is reached or when early stopping criteria are met.

Early-stopping combats overfitting by monitoring the model's performance on a validation set. A vali-

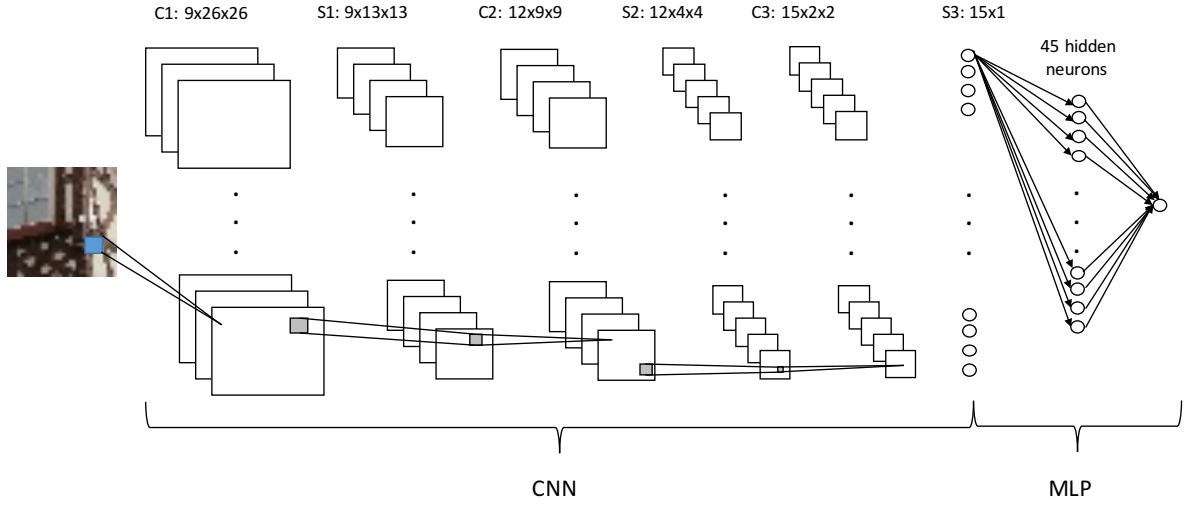

Figure 4: Deep learning model architecture.

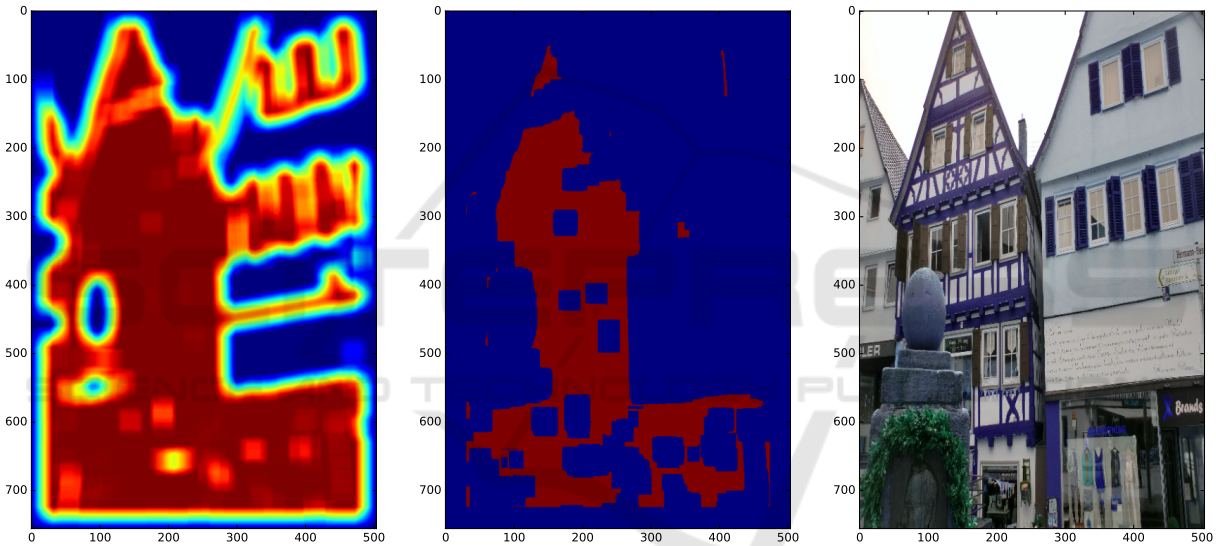

Figure 5: (left) Output of the classifier, (middle) thresholded output and (right) original image.

dation set is a set of examples that we never use for stochastic gradient descent, but which is also not a part of the test set. The validation examples are considered to be representative of future test examples. We can use them during training because they are not part of the test set. If the model's performance ceases to improve sufficiently on the validation set, or even degrades with further optimization, then a decision mechanism gives up on much further optimization.

Let us denote as $l_v^{(i)}$ the validation loss at the i^{th} training epoch and as $l_{v,i}^* = \min_{j=1\ldots i} l_v^{(j)}$ the minimum validation error till the i^{th} training epoch. For deciding when the optimization process must end we compute the quantity $p^{(i)}$ for each training epoch i. If after the i^{th} training epoch $l_v^{(i)} < l_{v,i-1}^*$ then $p^{(i)} = \max\{p^{(i-1)}, 2i\}$, otherwise $p^{(i)} = p^{(i-1)}$. The initial value $p^{(0)}$ can be set to the number of training epochs that we want the optimization algorithm to go through regardless the evolution of the validation error. When the quantity $p^{(i)} < i$ the decision mechanism terminates the optimization process.

The quantity p can be seen as the "patience" of the decision mechanism before terminating the optimization process. During the first training epochs the mechanism has a small amount of "patience", since the validation error is expected to be decreased fast enough. On the contrary, during later iterations the amount of "patience" is getting larger, since the improvement of validation error is slower due to the convergence properties of the optimization algorithm.

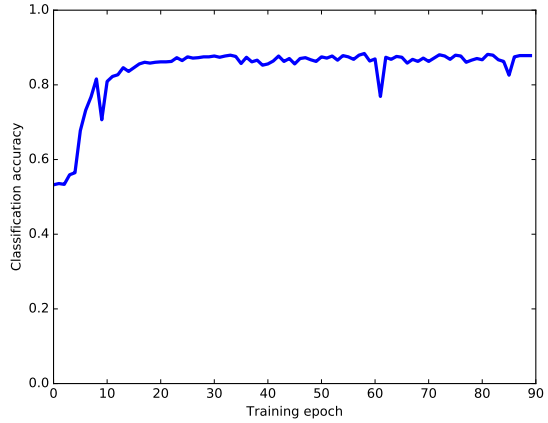

Figure 6: Classification accuracy versus the number of training epochs.

5.3 Building Recognition

After the training of the network, the proposed system is able to recognize half-timbered framed buildings on newly captured images. A new image is fed as input to our system. Initially, this image is being split into overlapping patches of 32×32 size. Each one of the patches is classified by the deep learning model as depicting or not some part of the building of interest. Then, the label of the patch is assigned to every pixel that belongs to it.

Due to the fact that we use overlapping windows, most of the image pixels belong to more than one patches and thus multiple labels are being assigned to them. For this reason, the final label of each pixel is determined by utilizing a voting mechanism that uses the average of the labels that have been assigned to a specific pixel.

6 EXPERIMENTAL RESULTS

Experimental results were obtained by creating a dataset that contains 3000 negative and 3000 positive samples. For training and evaluating the performance of the proposed approach, we split the dataset into train, test and validation sets, with splitting ratio $0.7 : 0.15 : 0.15$. Finally, we set the maximum number of training epochs to be equal to 400, while the initial value for p was set equal to 50.

Fig. 6 presents the classification accuracy on test set versus the number of training epochs. The training phase is terminated after 90 epochs due to the employment of early stopping criteria, which are used to avoid overfitting. Furthermore, due to early stopping the computational requirements of the training phase are being reduced.

After the training phase our learning model is able to achieve around 85% accuracy in terms of patch classification on the testing set of patches. At this point it has to be mentioned that, although the facades of half-timbered framed buildings have the same kind of patterns, their color and size may be different. The ability of our proposed architecture to achieve around 85% classification accuracy on the test set, implies that it can successfully generalize to unknown data. Fig. 5 presents a visualization of the results regarding the recognition of half-timbered framed buildings. The color in the output of the classifier, Fig. 5(left), corresponds to the probability a pixel to depict some part of the object of interest. Blue color corresponds to low probability, while red color corresponds to high probability. The output of the classifier was thresholded, Fig. 5(middle), in order to select the pixels that depict some part of the object of interest with probability at least 95%. As it can be seen, the proposed approach is able to discriminate half-timbered framed buildings than other types of buildings.

7 CONCLUSIONS

In this paper we proposed a deep learning based approach for recognizing half-timbered buildings in Calw city. The proposed approach acts as a proof of concept that deep learning based approaches can be used for recognizing specific types of urban buildings without relying on sophisticated construction of handcrafted features. Furthermore, although the training phase can be computationally expensive, due to the feed forward nature of CNNs and MLPs the proposed system can detect and recognize building very efficiently with low computational cost.

Finally, among the future perspectives is (i) the expansion of the developed framework for the detection and recognition of multiple building types and (ii) a thorough investigation on the effects of the network parameters on the classification accuracy.

ACKNOWLEDGEMENTS

This work is supported by the H2020 project: Transforming Intangible Folkloric Performing Arts into Tangible Choreographic digital Objects (TERPSICHORE) funded by European Union under the grant agreement 691218.

REFERENCES

Ali, H., Paar, G., and Paletta, L. (2007a). Semantic indexing for visual recognition of buildings. In *5th Int. Symp. on Mobile Mapping Technology*. Citeseer.

Ali, H., Seifert, C., Jindal, N., Paletta, L., and Paar, G. (2007b). Window detection in facades. In *Image Analysis and Processing, 2007. ICIAP 2007. 14th International Conference on*, pages 837–842. IEEE.

Baatz, G., Köser, K., Chen, D., Grzeszczuk, R., and Pollefeys, M. (2010). Handling urban location recognition as a 2d homothetic problem. In *European Conference on Computer Vision*, pages 266–279. Springer.

Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H. (2007). Greedy layer-wise training of deep networks. In *Advances in NIPS 19*, pages 153–160.

Bres, S. and Tellez, B. (2006). Localisation and augmented reality for mobile applications in culture heritage. *Lyon: INSA*.

Friedman, J. H., Bentley, J. L., and Finkel, R. A. (1977). An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software (TOMS)*, 3(3):209–226.

Hinton, G., Osindero, S., and Teh, Y. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554.

Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110.

Makantasis, K., Karantzalos, K., Doulamis, A., and Loupos, K. (2015). Deep learning-based man-made object detection from hyperspectral data. In *International Symposium on Visual Computing*, pages 717–727. Springer.

Mnih, V. and Hinton, G. E. (2012). Learning to label aerial images from noisy data. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012,*. icml.cc / Omnipress.

Montavon, G., Orr, G., and Muller, K.-R. (2012). *Neural Networks: Tricks of the Trade*. Springer Berlin Heidelberg.

Müller, P., Zeng, G., Wonka, P., and Van Gool, L. (2007). Image-based procedural modeling of facades. *ACM Transactions on Graphics (TOG)*, 26(3):85.

Olojede, A. and Suleman, H. (2015). Investigating image processing algorithms for navigating cultural heritage spaces using mobile devices. In *International Conference on Asian Digital Libraries*, pages 215–224. Springer.

Schindler, G., Krishnamurthy, P., Lublinerman, R., Liu, Y., and Dellaert, F. (2008). Detecting and matching repeated patterns for automatic geo-tagging in urban environments. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–7. IEEE.

Shechtman, E. and Irani, M. (2007). Matching local self-similarities across images and videos. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE.

Singhal, A., Luo, J., and Zhu, W. (2003). Probabilistic spatial context models for scene content understanding. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–235. IEEE.

Tell, D. and Carlsson, S. (2000). Wide baseline point matching using affine invariants computed from intensity profiles. In *European Conference on Computer Vision*, pages 814–828. Springer.

Tell, D. and Carlsson, S. (2002). Combining appearance and topology for wide baseline matching. In *European Conference on Computer Vision*, pages 68–81. Springer.

Wendel, A., Donoser, M., and Bischof, H. (2010). Unsupervised facade segmentation using repetitive patterns. In *Joint Pattern Recognition Symposium*, pages 51–60. Springer.

Xiao, J., Fang, T., Zhao, P., Lhuillier, M., and Quan, L. (2009). Image-based street-side city modeling. *ACM Transactions on Graphics (TOG)*, 28(5):114.