Classification of Masonry Bricks using Convolutional Neural Networks – a Case Study in a University-Industry Collaboration Project

Mika Iitti, Jere Grönman, Jari Turunen and Tarmo Lipping

Tampere University/Faculty of Information Technology and Communication Sciences, Pori, Finland

mika.iitti@tuni.fi

Abstract – **This paper presents a case study – developing a computer-based classification framework to classify masonry bricks into three quality categories – carried out as a part of the Robocoast R&D Center project. The project aims at better collaboration between universities and industry by establishing an innovation platform where companies can bring their challenges to be addressed together with university experts. The project also promotes collaboration between universities being a part of the RoboAI Competence Centre – a joint research and innovation platform of Satakunta University of Applied Sciences (SAMK) and Tampere University, Pori unit. Automatic classification of bricks is important as it is foreseen that a robotic arm, powered by an automatic classifier, could replace the heavy and tedious work currently performed by humans in brick factories. A convolutional neural network-based solution, using a pretrained VGG-16 deep learning architecture, is proposed. Overall accuracy of 88 % was obtained when considering all three quality classes. When only discarding class 3 bricks, i.e., those that are not suitable for any construction work, the accuracy was 93 %.**

Keywords—university-industry collaboration, classification, data acquisition, CNN, transfer learning

I. INTRODUCTION

Opening a dialogue between university and industry can sometimes be challenging. Both parties are driven by slightly different motives and goals. A solution to this has been sought in the Robocoast R&D Center project, which aims to create a better connection between academia and industry. A major aspect of the collaboration is to provide companies with an innovation platform where they can present their challenges and together with university experts, a possible solution to the challenge is found. Another important part of the project is to establish an innovation platform for collaboration between universities in the region. A RoboAI innovation platform for research and development as well as education and competence has been established between Satakunta University of Applied Sciences (SAMK) and Tampere University Pori unit. RoboAI is also a competence center of the Robocoast Digital Innovation Hub. There are a few objectives that the innovation platform offers to the region's industry:

- to support the growth of technology industry and enterprises in the region

- to increase competitiveness and create a solid basis for industrial digitalization

- to offer a common research infrastructure and high-quality research and international product development platform for the companies to use.

The project carried out various pilot studies together with industry. This paper presents a case study conducted by our research group of Data Analytics and Optimization (DAO) together with an industrial partner. Our research group has versatile competence in the different sectors of data analytics and decades of experience in applying data analytics for the needs of science and industry.

This paper presents a challenge brought up by an industrial company to which a solution is developed. We first describe the problem of masonry brick classification on a brick factory production line. We then present the data acquisition framework used to obtain the data for training the deep learning algorithms. After that, the solution employing the pretrained VGG-16 network architecture in presented, the classification results are given and some issues to be considered when the developed framework will be used in the production environment are discussed.

II. BRICK QUALITY ASSESSMENT

Bricks are the oldest known building material to humans. Bricks and brickmaking dates to 7000 BC in southern Turkey and in the ancient settlement near the city of Jericho. Industrial revolution after 1885 transformed brickmaking from handwork to machinery-based production [1]. Nowadays large-scale brick production is fully automated, including robotics as described in [2].

Brick quality is closely related to the production conditions in the factory such as humidity, temperature in various compartments and its control, modernity of the machinery etc. These conditions become more importing in small production units. Raw material properties should be as uniform as possible,

as well as the mixing, drying, firing and cooling phases in masonry brick industry. The environmental conditions can be monitored using sensor networks to get feedback for quality improvement. In [3], for example, a cost-effective PLC based sensor / monitoring system has been tested in a small masonry brick plant.

Brick quality is usually inspected manually in smaller plants when sorting the bricks to separate storage platforms such as pallets, for example. It is easy to classify the bricks if the bricks are labelled to broken or not broken. However, if there are more classes such as 1st quality (flawless), 2nd quality (minor issues) and 3rd quality (broken or severely deformed), then the classification requires expertise and trained eye. In [4], an image-based crack detection system of bricks using conventional signal processing methods is presented.

When reviewing research papers on brick classification, a common task considered seems to be that of brick recognition from the brick walls using AI [5,6]. In [5], the degradation aspects of mortar and bricks are searched from the image to evaluate the repairment need of the brick wall. In [6] brick identification is extended to rocks and ashlar and damage identification also includes biofilms etc.

In this paper an automatic brick classification algorithm is developed to be used in a brick factory to control a robotic arm to collect 1st and 2nd class bricks to pallets and leave 3rd class bricks to conveyor belt for crushing.

III. DATA ACQUISITION ENVIRONMENT

A. Material

In this case study we tested brick classification into three categories using neural networks. The first class is flawless, good quality bricks while the second class contains bricks with minor defects, such as minor cracks between holes, minor chipping, color changes etc. These bricks can be used for construction in less critical sites. The third class contains broken, bent, miscolored or otherwise unsuitable bricks for construction. The brick belonging to the third class are abandoned. We obtained a set of tiles, consisting of 2x128 tiles of the 1st class, 2x128 tiles of the 2nd class and 5x128 tiles of the 3rd class, as shown in Figure 1.

Fig. 1. Two left pallets filled with 3rd class tiles, right side pallet stacked with 2nd class tiles (9 lower rows) and 1st class tiles on top.

B. Data acquisition

The tiles were imaged using the IDS area camera, model UI-3590CP-C-HQ. The camera was placed inside the dome with led lighting over the conveyor belt. The conveyor belt was imitating the conveyor belt in the brick factory. The dome height was adjusted in aluminum frame to match the optimal imaging conditions. The camera was controlled by Halcon machine vision software. When the software identifies the rough shape of a detected brick and the color in certain limits to trigger the camera, the image of the brick is captured automatically. The image capturing setup is shown in Figure 2.

Fig. 2. Image acquisition setup

C. Data preprocessing

Each brick was photographed from three sides, top side, bottom side, and the outer surface side in constant light conditions. Each image triplet was assembled as one image. An example of a brick image triplet is presented in Figure 3.
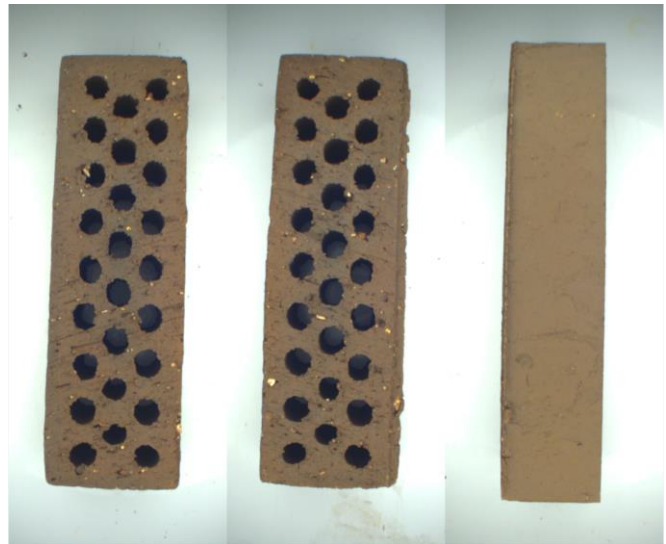

Fig. 3. A first class brick image triplet.

In Figure 4, two images of bricks classified manually into class 2 and three images of bricks belonging to class three are presented. Color variation and minor chipping can be observed when visually examining the 2nd class bricks. The bricks belonging to the third class have major defects or distorted shape.
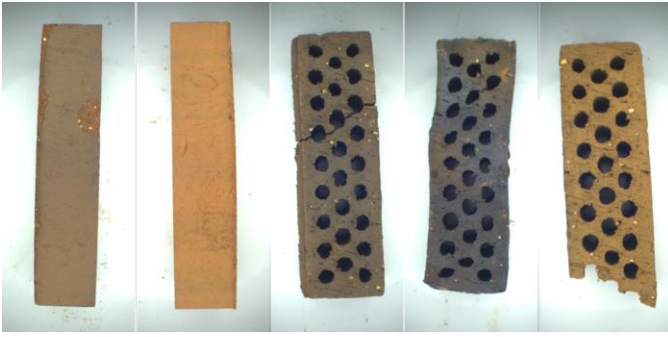
Fig. 4. Brick belonging to class 2 due to minor defects are shown in two lettmost panles. Three rightmost images present abandoned tiles that belong to class three.

Total number of triplet-images are shown in Table 1.

TABLE I. TOTAL NUMBER OF TRIPLET IMAGES

| Class | Triplet images for training | | Triplet images reserved for testing |
	Number of images	Number of images after oversampling	
1st class	256	297	128
2nd class	256	297	128
3rd class	593	297	296

The number of images in class 3 is larger than in classes 1 and 2 combined. To avoid class unbalance in training the neural network, the image sets of classes 1 and 2 were oversampled randomly by 2.3 times to match the size of class three dataset. The oversampling was done only for network training image triplets.

IV. BRICK CLASSIFICATION USING CONVOLUTIONAL NEURAL NETWORK

A. Initial experiments and transfer learning with VGG-16

Convolutional neural networks (CNN) are based on the early works of Yann Le Cun. He used the backpropagation algorithms for training the convolutional neural network to recognize the handwritten digits of the area codes for US Postal Service [7]. During several years of research, Le Cun and colleagues reviewed various methods for handwritten digit recognition and the results showed that the CNN outperformed all other models [8]. However, the general CNN architecture requires large amounts of training data.

The VGG-16 network architecture [9] was originally developed for ImageNet Large-Scale Visual Recognition Challenge (ILSVRC-2014). The ILSVRC-2014 dataset consists of over 14 million images belonging to roughly 1000 classes. For the competition a subset of the whole image database, containing 1281167 training images, 50000 validation images and 100000 test images belonging to 1000 classes was used [10]. Recognition results of the VGG-16 were comparable with the ILSVRC-2013 winner GoogLeNet with 92.7% accuracy in top-5 test [9]. VGG-16 network consists of several layers. The image input size is 224 x 224 x 3 and it is passed through several 3 x 3

small receptive field filtering layers. Spatial padding of 1 pixel is added to preserve spatial resolution in convolution layers. Max pooling is performed using 2 x 2 -pixel windows with stride = 2 [9]. After convolution layers the last image representation is fed to three layer (4096-4096-1000) fully connected network with 1000 outputs.

In our case VGG-16 network was taken to the tests as it was but the output layer was modified for the brick classification containing only three outputs in first test, as shown in Figure 5.
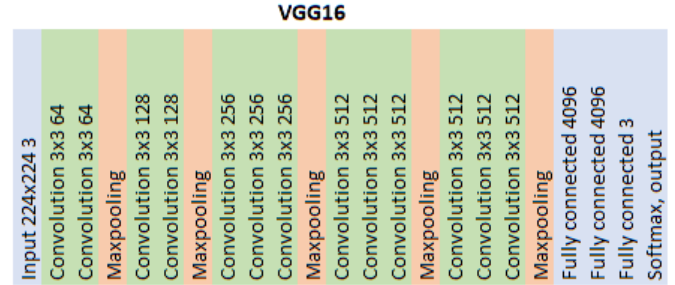

Fig. 5. The VGG-16 modified structure in the three class recognition test. Image is reproduced from [11]

In second test the output layer was modified for the brick classification containing only two outputs, as shown in Figure 6.
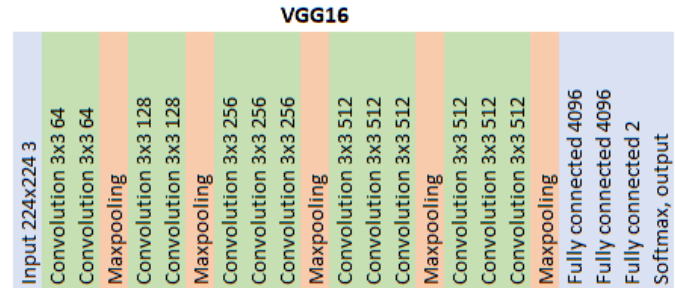

Fig. 6. The VGG-16 modified structure in the two class recognition test. Image is reproduced from [11]

As our brick image database was too small to adequately train a CNN capable of performing the classification task, it was decided to base the algorithm on a pretrained VGG-16 network and to use transfer learning. Pretraining was done using the ILSVRC-2014 dataset. In both tests fully connected 4096-4096 network layers were trained with the brick images to obtain brick classification results.

B. Training of the CNN

Brick triplet images were loaded to the classification algorithm from three different folders according to classes. Classes 1 and 2 were also randomly oversampled in this phase to match the size of class 3. The 50/50 cross-validation division was made so that the training and validation data sets contain 50 % of the data (25% each) and testing data contained the rest of the data. The data were organized in such a way that 149 images from each class were reserved for training, 148 images

for validation and 128 images from classes 1 and 2 and 296 from class 3 for testing. All the set selections were made

randomly. Transfer learning of the VGG-16 model was performed using 15 epochs. Each epoch contained 14 training steps and 14 validation steps using batch size of 32. The computation was done in Python Keras platform. Loss function used during tests was categorical cross-entropy and stochastic gradient descent SGD utilized as optimizer with fixed learning rate of 0.0001.

There were two cases is CNN training: Test 1 was to train and test the network to identify all three classes simultaneously. In test 2 the classes were trained and identified pairwise: class1 vs. class2, class1 vs. class3 and class2 vs. class3.

In both cases, the architecture and the hyperparameters as well as the data structure and organization were the same.

V. RESULTS

In test 1, the result of model training to classify all three classes is shown in Figure 7.
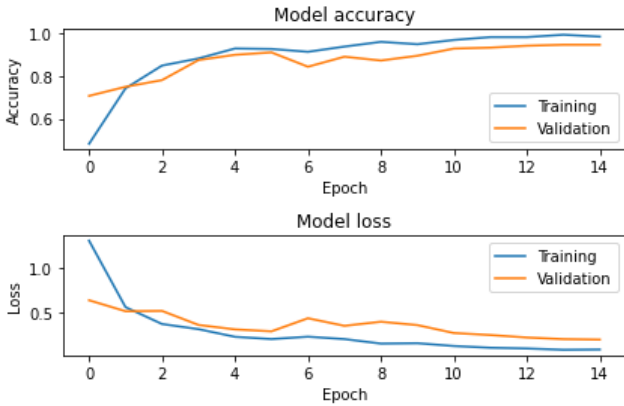

Fig. 7. Training and validation curves of VGG-16 in brick classification.

In first test training data accuracy was 99.0 %, validation accuracy 92.0% and test data accuracy was 88.0%. The confusion matrix is shown in Figure 8.
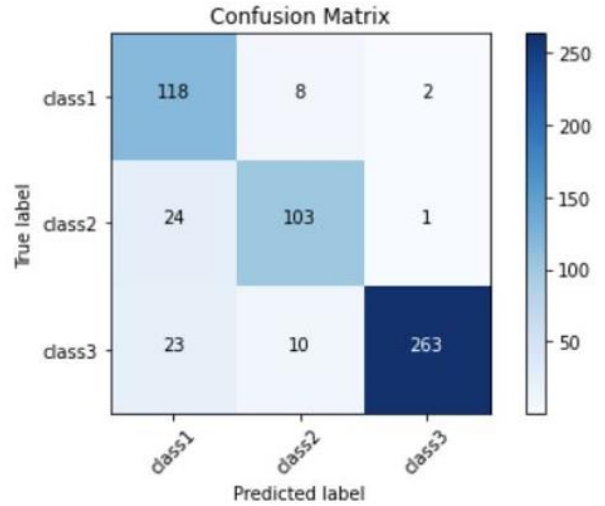

Fig. 8. Confusion matrix of brick classification using 128 (for classes 1 and 2) and 296 (class 3) test data images in test 1.

In test 2, the VGG-16 network architecture and the dataset division were the same as in the first test. The pairwise classification results are presented in Table 2.

The results are somewhat similar when comparing the test 1 and test 2 results. However, it is interesting to notice that the class1-class2 separation is more challenging when compared to class separation against class 3. Class 1 and class 2 bricks may be too close to each other near classification border cases and that may cause the wrong classification sometimes. The larger test set in class three should not affect to the results, because on both tests the training data sets were equal among three classes.

VI. DISCUSSION AND CONCLUSION

In this collaboration experiment it shows possibilities to utilize machine learning for industrial use. When we set out to develop a solution to the challenge posed by the industrial company, it was initially good to find out exactly the limits of the study. In initial phase, what type of brick would be selected for investigation. Second, how many bricks are needed to form a sufficient dataset. Third, which is sufficient recognition accuracy. And it was also important to keep in mind the follow-up to this study.

TABLE II. DIFFERENT ACCURACIES OBTAINED FROM PAIRWISE CLASSIFICATION

Case	Different accuracies obtained from pairwise classification					
	Training accuracy	SD	Validation accuracy	SD	Testing accuracy	SD
Class1-Class2	97.0 %	0.04	97.0 %	0.04	87.0 %	0.04
Class1-Class3	100 %	<0.01	93.0 %	0.02	92.0 %	0.02
Class2-Class3	100 %	<0.01	96.0 %	0.02	93.0 %	0.02

The overall quality of the recognition mechanism is good, but it could be better if there were more images of the three classes. The case study investigated how to classify bricks of different qualities based on cracks, colors or shape deformations. Only one type of brick was used in the study, which had factory requirements for shape and color. The results of the study were good and of such a quality that it is possible to move on to the next stage of development. The next step is to apply the identification method together with the robot solution, so that the robot can classify the bricks directly on the automation line and allocate them to the right pallets.

Before applying the identification method together with the robot solution on the automation line, the requirements of different types of bricks must be taken into account. Brick types can vary and there can be different colors and shapes. The speed of the identification method must also be optimized to support a robust solution that is efficient enough. One solution to this could be to apply the reinforcement learning method, which would make use of an existing implementation and be able to identify the classification of different types of bricks.

In reinforcement learning, there is a possibility to train the network online at the conveyor belt. When the bricks are passing the fixed camera station, an operator, near camera station, will decide the class information of the current brick and send this information to the network by pushing one of the pushbuttons that indicate one of the three classes, for example. With this information, network will gradually adapt to new information. This method can also be used for the totally different brick types.

Acknowledgment

References

[1] URL: https://brickarchitecture.com/about-brick/why-brick/the-history-of-bricks-brickmaking, accessed 13th Oct. 2021.

[2] C. Hanser, 1999. Fully Automated Masonry Plant. In proc. of 16th IAARC/IFAC/IEEE International Symposium in Automation and Robotics in Construction, Sept 22-24, 1999, Madrid, Spain, 295-300.

[3] S. Lahiri, 2016. Quality Control of an Automated Brick Manufacturing Plant. In International Journal of Innovative Research in Science, Engineering and Technology, 5 (13), Oct. 2016, 302-309.

[4] K.A. Schmitt, J.R. Riddington, R.C.D. Young, D.M. Budgett, C.R. Chatwin, 2000. Image Processing Applied to Brick Quality Control. In The International Journal of Advanced Manufacturing Technology, 16, 2000, 434-440.

[5] R. Kajatin, L. Nalpantidis, 2020. Image Segmentation of Bricks in Masonry Wall Using a Fusion of Machine Learning Algorithms. In proceedings of ICPR 2020 workshop on Pattern Recognition in Construction and the Built Environment, January 10, 2021, Milan, Italy, 1-16.

[6] Y. Ibrahim, B. Nagy, C. Benedek, 2020. Deep Learning-Based Masonry Wall Image Analysis. In Remote Sensing, 12, 2020, 3198, 1-28.

[7] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, 1989. Backpropagation applied to handwritten zip code recognition. In Neural Computation, 1 (4), 541-551.

[8] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, 1998. Gradient-based learning applied to document recognition. In Proceedings of the IEEE. 86 (11), 2278 - 2324.

[9] K. Simonyan, A. Zisserman, 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv:1409.1556v6 [cs.CV], 1-14.

[10] ImageNet database. URL: https://www.image-net.org/download.php . Accessed October 18, 2021.

[11] Neurohive VGG-16, URL: https://neurohive.io/en/popular-networks/vgg16/ . Accessed October 18, 2021.