Classifiers Ensemble of Transfer Learning for Improved Drill Wear Classification Using Convolutional Neural Network

Jarosław Kurek[1], Izabella Antoniuk[1], Jarosław Górski[2], Albina Jegorowa[2], Bartosz Świderski[1], Michał Kruk[1], Grzegorz Wieczorek[1], Jakub Pach[1], Arkadiusz Orłowski[1], Joanna Aleksiejuk-Gawron[3]

[1]*Institute of Information Technology*

[2]*Institute of Wood Sciences and Furniture*

[3]*Institute of Mechanical Engineering,*

Warsaw University of Life Sciences – SGGW, Warsaw, Poland

jaroslaw_kurek@sggw.pl

Abstract. In this paper we introduce the enhanced drill wear recognition method, based on classifiers ensemble, obtained using transfer learning and data augmentation methods. Red, green and yellow classes are used to describe the current drill state. The first one corresponds to the case when drill should be immediately replaced. The second one denotes a tool that is still in a good condition. The final class refers to the case when a drill is suspected of being worn out, and a human expert evaluation would be required. The proposed algorithm uses three different, pretrained network models and adjusts them to the drill wear classification problem. To ensure satisfactory results, each of the methods used was required to achieve accuracy above 90% for the given classification task. Final evaluation is achieved by voting of all three classifiers. Since the initial data set was small (242 instances), the data augmentation method was used to artificially increase the total number of drill hole images. The experiments performed confirmed that the presented approach can achieve high accuracy, even with such a limited set of training data.

Key words: classifiers ensemble, convolutional neural networks, data augmentation, deep learning, tool condition monitoring.

1. Introduction

Manufacturing furniture can be lengthy and complex process, where each decision can result either in good quality product or large losses in income for the company, when entire piece or its elements do not meet quality requirements. One of many factors that can influence final outcome is drill sharpness, and pointing out exact moment when tool should be replaced to avoid budget losses due to poor product quality. Since manual observation of the drill state is not efficient enough, researching solutions that can automatize this stage was necessary.

Usually when tool wear problem is approached three classes would be used to describe its state: red, green and yellow. If evaluated tool would be assigned to first class, it would indicate that it already is in a poor state, and should be replaced immediately, since its further use in production process can generate loss for the company. Contrary, second class would denote tools that are still in good shape, and can be further used without any risk. Final class contains tools that are suspected of being to worn out, and therefore requiring human expert to manually evaluate their state – depending on given opinion, they can be either discarded or used further in production process.

Tool condition monitoring problem (or TCM) is not a new concept, similarly as evaluating drill condition specifically. When it comes to the latter, any described solutions usually required vast collections of different sensors for signal measurement. Collected signal data will then be processed and used to generate diagnostic features for each drill state. Commonly used signals can be related to: feed force, noise, cutting torque, vibration or acoustic emission [4]. While above setups can render accurate classifications, to achieve acceptable results usually numerous preprocessing stages will be required, including steps such as appropriate sensor or signal choice, generating and selecting best diagnostic features or building the classification model for given set of input data. Because of that, solutions from that set are quite complex, require lengthy preparations, and usually will be rather expensive due to various equipment parts required to even start measuring required signals. Moreover if later on some of selected input elements would prove unnecessary or inadequate, it would automatically generate loss (in terms of equipment required for measurements and time needed for setup configuration), and any mistakes during early stages may result in final product being not accurate enough. Additionally, in [1,2,3] even though authors took into account various features generated from different signals, accuracy of presented procedures did not exceed 90% for given task of recognizing three drill wear classes.

One of the goals of the presented approach (which should be treated as a continuation of the research presented in [5] and in our other paper in this volume [6]) is finding a more efficient and less complicated way (especially in terms of required equipment) for achieving the same classification task, with acceptable accuracy, exceeding the 90% threshold. Similarly as in [5, 6] the images of drilled holes will be used as a base for the assessment process. Since in the previous tests the transfer learning proved to give satisfactory results, it will also be used in the current approach, but instead of using a single model, three of the most commonly used pretrained convolutional neural networks (CNN) [7, 10] will be adjusted to our classification problem and used in the final classifier ensemble. Since the initial training set is rather small (242 images, representing three classes: 102 samples for green, 60 samples for yellow and 80 samples for red drills), our solution also includes the data augmentation method, to artificially increase the number of examples, similarly as in [7] and [6].

In this paper an approach is proposed that includes the creation of the classifier ensemble, composed of three pretrained convolutional neural networks. Out of the available

(a) Green (b) Yellow (c) Red

Fig. 1: Examples of holes produced by each drill wear class: green (a), yellow (b) and red (c).

solutions we chose AlexNet, VGG19 and VGG16 and adjusted them to our classification problem using transfer learning and data augmentation methodologies (with additional requirement that each individual model will achieve accuracy above 90%). The final result is achieved by counting votes from all the classifiers. Similarly as in previous approaches to this subject, we are using three classes to describe the drill wear state, while the initial image set used for training and evaluation is the same as in [5, 6].

2. Data augmentation methodology

Data samples used in our experiments were collected in cooperation with Faculty of Wood Technology at Warsaw University of Life Sciences, using standard Buselatto JET 100 CNC vertical machining centre. For the test purposes drilling process was performed on standard laminated chipboard (Kornopol U 511 SM) that is typically used in furniture industry. Dimensions of the test piece were $150 \times 35 \times 18$ mm. Regular 12 mm FABA drill equipped with a tungsten carbide tip was used for the drilling process.

Data set that is used for CNN learning has the same structure as the one presented in [5]. From 242 examples present in initial set, we have the following attribution for each of the three classes: 102 images for green, 60 images for yellow and 80 images for red. Examples representing each of the defined classes are shown in Fig. 1 (images in this Figure and in Fig. 2 are similar to those used in our previous papers on drill wear classification due to that we used the same initial set of images, and the image transformations used in [6]).

Since the initial set is not sufficient for most of deep learning approaches, we decided to artificially expand it, using basic image operations. 18 different operations were used

Tab. 1: Parameters of the pretrained convolutional neural networks chosen for classifier ensemble.

CNN	**Depth**	**Size [MB]**	**Parameters [$\times 10^6$]**	**Input Image Size**
AlexNet	8	227	61.0	227×227
VGG16	16	515	138	224×224
VGG19	19	535	144	224×224

for each of the original images from the initial data set. After performing those our training data set consisted of total 4598 examples, where 1938 samples were assigned to green class, 1140 samples to yellow class and 1520 samples for red class. Image operations that were used for expanding the original data set are as follows:

1. ColorToGrayscale – convert image to gray-scale values.
2. ColorBrightJitter – adjust brightness of image by random offset in range [-0.3, -0.1].
3. ColorContrast1 – adjust contrast of image by scale factor in range [1.2, 1.4].
4. ColorContrast2 – adjust contrast of image by scale factor in range [1.4, 1.6].
5. ColorHueJitter1 – change image hue by random offset in range [0.05, 0.15].
6. ColorHueJitter2 – change image hue by random offset in range [0.15, 0.30].
7. NoiseGauss – add Gaussian noise to the image.
8. Noise1 – add salt and pepper noise to the image, with strength factor 0.2
9. Noise2 – add salt and pepper noise to the image, with strength factor 0.4
10. Reflection – create a reflection that transforms original image by flipping it in each dimension with 50% probability.
11. Rotate30 – rotate image by random angle in range [0, 30].
12. Rotate45 – rotate image by random angle in range [30, 60].
13. Rotate90 – rotate image by random angle in range [60, 90].
14. Rotate120 – rotate image by random angle in range [90, 120].
15. Scale1 – scale image by random factor in range [1.2, 1.5]
16. Scale2 – scale image by random factor in range [0.8, 0.9]
17. Shear – shear transformation (horizontal), with angle selected randomly in [-30, 30].
18. Translate50 – translate image both vertically and horizontally by random number of pixels in range [-50,50].

The final operation that was required for artificially augmented data set was the image size adjustment, since each of pretrained convolutional neural networks that was chosen for our approach requires images to be saved with specified dimensions. Example results for each of denoted image transformations are presented at Fig. 2. Specific description of each of those operations are in [20]. Parameters of selected models are presented in Table 1.

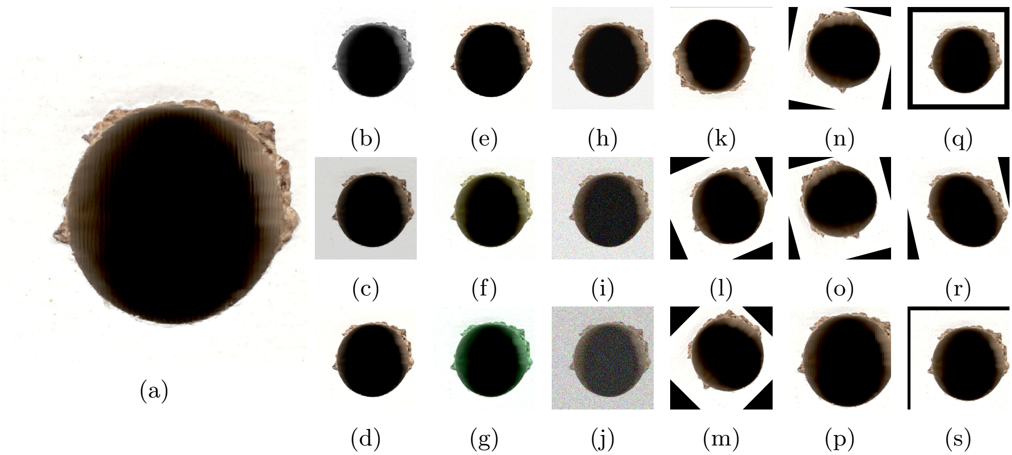

Fig. 2: Examples of augmented data set for a single original image (**a**): (**b**) ColorToGrayscale, (**c**) ColorBrightJitter, (**d**) ColorContrast1, (**e**) ColorContrast2, (**f**) ColorHueJitter1, (**g**) ColorHueJitter2, (**h**) NoiseGauss, (**i**) Noise1, (**j**) Noise2, (**k**) Reflection, (**l**) Rotate30, (**m**) Rotate45, (**n**) Rotate90, (**o**) Rotate120, (**p**) Scale1, (**q**) Scale2, (**r**) Shear, (**s**) Translate50

3. Classical deep learning approach (CNN)

When it comes to classification problems, in recent years one of most commonly used solutions (especially if different types of images are considered) are different deep learning approaches [8, 9, 10, 11]. Among those CNN (convolutional neural network) would be typically used either as a main function, or as one of the core parts of entire approach. Each of hidden layers of nonlinear processing would begin extracting diagnostic features (starting from elements such as points, edges, corners, etc.), while each successive layer will slowly derive higher level features from the more basic ones, creating hierarchical data representation of input image. Thanks to that there is no need for the user to manually specify diagnostic features for given problem, as they will be obtained during training process. Additionally, features found in such a way tend to be more universal and usually give good results for different types of images, with acceptable accuracy.

Deep learning approach uses blocks of pixels with fixed size for analysis (i.e. 5 by 5, 10 by 10 or in case of larger images 15 by 15 pixels), contrary to regular methods which iterate through image pixels. In this approach few different filtrations are used to achieve final results first starting with linear filtration. In the next step nonlinear transformation is applied to filter output signals (usually rectified linear unit or ReLU). Finally, reducing size of the actually processed blocks, pooling operation is performed.

Main difference between deep learning and classical approach lies in diagnostic features extraction. When classical machine learning is considered this process is considered as separate step, usually consisting of different stages, often complicated in themselves. In case of deep learning all features are generated automatically as an embedded internal process that is done in hidden layers. Extracted features can later be used as attributes for external classifier (i.e. support vector machine) or input values for softmax classifier (which is an integral part of CNN). For detailed description of CNN solution used as reference in this work see [5].

4. Transfer learning approach (CNN AlexNet)

Database used in our approach consists of 242 images for three defined classes (102 for green, 60 for yellow and 80 for red class). Such number of training examples is not sufficient enough for training CNN from the scratch. Therefore, in order to ensure better classification accuracy, using model pretrained on much larger set of different and unrelated images was one of possible solutions. Using transfer learning methodology [13, 14], AlexNet model, created by Krizhevsky, Sutskever and Hinton [12, 13, 14] and prepared with Matlab [15], was applied to our classification problem. AlexNet was pretrained using more than a million images, where each of them represented one of 1000 defined classes [15, 16].

In our approach structure of used pretrained CNN is made of 9 layers, containing 25 sublayers. Softmax classifier, which is part of this solution, calculates i-th component of output vector (which represents given number of classes M) by inserting u-vector in the following equation:

$$\mathrm{softmax}(u)_i = \frac{\exp(u_i)}{\sum_{j=1}^{m} \exp(u_j)} \tag{1}$$

Softmax function uses exponential normalization and its values range from 0 to 1. Final classifier value is treated as possibility of specified class, while component with highest value will indicate chosen class. To calculate softmax function for our case of three classes (M=3), cross entropy has been used.

AlexNet requires for all images to have the same size, which is 227 by 227 by 3, so all input images used in experiments were additionally adjusted to meet those specifications. In tested solution from all trained layers, first six were applied without any changes and implemented the same mechanism for creating diagnostic features as in classes of images used in pretraining stage of AlexNet. The adjustments were made in last three, fully connected layers, to better adapt entire setup for our specific classification problem.

5. Classifier ensemble

In the previous work [5] the pretrained AlexNet CNN model achieved promising results (see Table 2), but was clearly losing in terms of training data. Therefore, we decided to expand the data set in our current approach. Since even without the need to use specialized equipment in the data acquisition process, collecting samples of drill hole images could take a considerable amount of time, it was decided that artificial data augmentation will be used, with basic image operations. This approach would also ensure that user needs to take minimal number of actions in order to achieve acceptable accuracy of final classifier. Total of 18 operations for different types of image transformations were chosen (like rotation, adding noise, scaling etc.), and all elements from initial data set were processed through each of them, with random values for most operations. Such approach ensured maximal diversity in final set, without adding results that would be too different from accepted standards.

To further increase overall performance of our solution, instead of using single classifier we decided to apply classifier ensemble. We chose three of the most popular, pretrained CNN networks: AlexNet, VGG16 and VGG19. Taking into account previous experiments and CNN general properties, we assumed that this kind of approach will heighten final algorithm accuracy, without the need to use more complex solutions (like adding SVM as a final classifier as it was done in [5]) or significantly increasing calculation time (as with classical CNN, without using transfer learning methodology).

Each of the classifiers was prepared separately, while final class recognition is done by all three classifiers voting. For our approach, class that got most votes is the one that is assigned to presented image. For overview of learning process for each of chosen classifiers refer to Fig. 3.

6. Results

Initial data set prepared for the training and experiment purposes contained 242 images that represented three defined classes. After data augmentation process was performed, the set was extended to 4598 images, where each of new images was generated from image in original set, using basic image operations (see Fig. 2).

Due to different algorithms used to evaluate final solution, we use few methodologies to prepare and evaluate out experiments. For the first two algorithms only initial data set was used (242 images). This set was randomly divided into 10 subsets, and in further experiments 9 of those were used in training process, while remaining set was used for testing. We repeated those experiments 10 times in 10-fold cross validation mode, with test data being exchanged for each subsequent run. For three pretrained classifiers, using artificially extended data set, we used similar approach, but in that case 95% of data was assigned as training set, and remaining 5% was used for testing purposes. For classifier

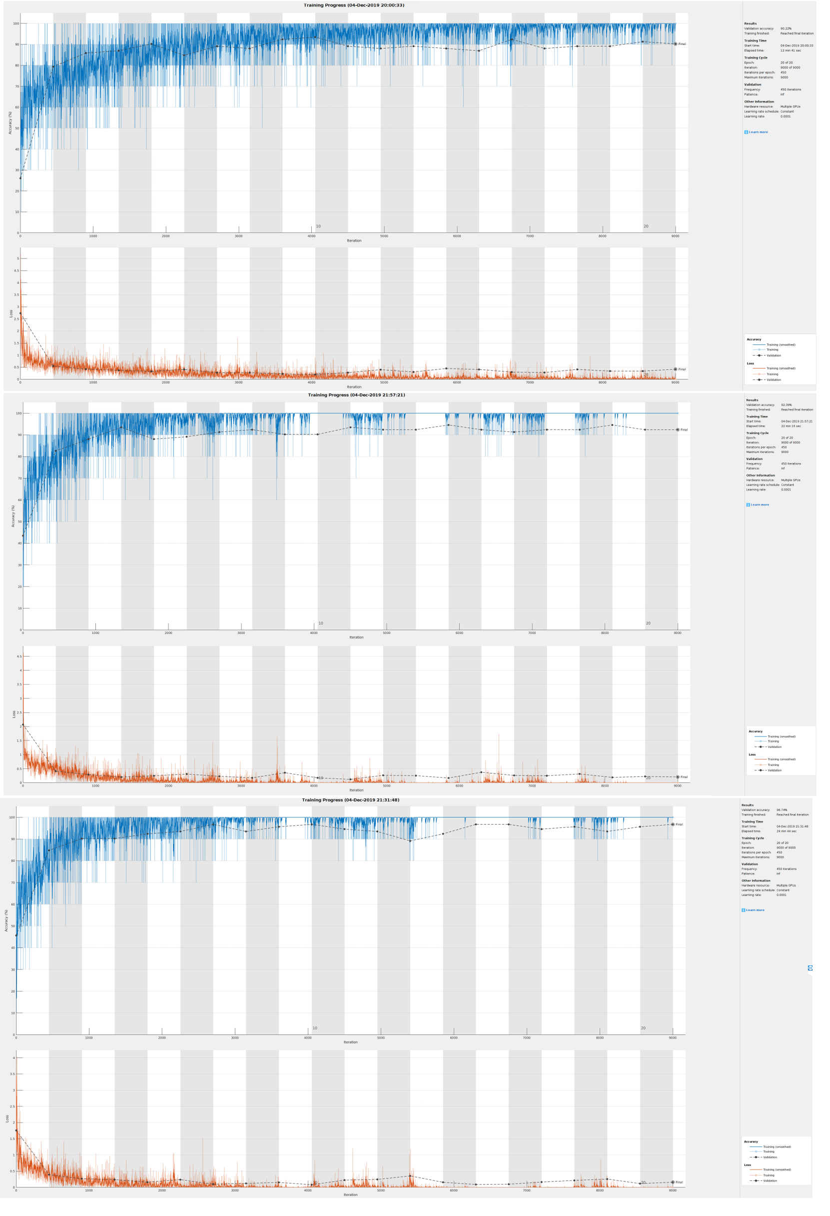

Fig. 3: Outline of training process for CNN models with augmented data set: AlexNet (top), VGG16 (middle) and VGG19 (bottom).

Tab. 2: Accuracy of chosen algorithms applied to the problem of drill condition classification with 3 classes (green, yellow and red). First two algorithms use 242 input images, while rest of the algorithms used artificially expanded set of 4598 images.

No.	**Deep learning algorithm**	**Accuracy[%]**
(1)	Standard CNN	35%
(2)	Pretrained CNN	85%
(3)	AlexNet CNN	94.30%
(4)	VGG16 CNN	92.39%
(5)	VGG19 CNN	96.73%
(6)	Classifier ensemble	95.65%

ensemble solution, trained classifiers were voting for final class assignment. Obtained results are prepared in the form of mean value of class recognition accuracy and are presented in Table. 2. Subsequent table rows refer to:

1. **Standard CNN** – CNN which learned from scratch.
2. **Pretrained CNN** – CNN using pretrained AlexNet model with stoftmax classifier.
3. **AlexNet** – pretrained AlexNet CNN, using augmented dataset.
4. **VGG16** – pretrained VGG16 CNN, using augmented dataset.
5. **VGG19** – pretrained VGG19 CNN, using augmented dataset.
6. **Classifier ensemble** – solution using AlexNet, VGG16 and VGG19 models, adjusted to presented classification problem with classifier voting as a method to select final class assignment.

As in previous work [5] it is clearly visible that using pretrained CNN for such a small dataset has a great advantage. Moreover, by artificially expanding the data set we were able to successfully imitate a larger number of samples, and to increase the overall accuracy both for single classifiers (AlexNet – 94.30% accuracy, VGG16 – 92.39% accuracy and VGG19 – 96.73% accuracy) and for the final solution, combining votes of all three of them (95.65% accuracy). Even though VGG19 classifier achieved higher accuracy than the final solution, the presented approach still achieved more than acceptable outcomes and has grater generalization capabilities.

7. Conclusion

In this paper we presented a method for drill wear state recognition, based on drilled holes. Our approach improves on previous solutions [5] by using classifier ensemble of three most popular, pretrained convolutional neural networks (AlexNet, VGG16 and VGG19), bringing into play transfer learning and data augmentation methodologies to further improve final class recognition accuracy. Previously to increase precision of given classifications required either more complex solutions (like using SVM as a final classification method), or significantly larger data set used for learning (which could result in prolonged computations, like it was the case with CNN learning from scratch in [7], where similar accuracy of 95.5% was achieved for 33300 images, with training time that lasted over 20 hours).

Obtained results confirm that presented approach was able to achieve satisfying accuracy while predicting drill wear state. Outcome of classifier ensemble solution was compared both to traditionally trained CNN and previous version of current method, using pretrained AlexNet without data augmentation techniques, and accuracy rate was better in both cases. Additionally final algorithm was able to achieve higher precision than solution using SVM as a final classifier (see [5]). When it comes to training time, none of the classifiers in final solution required more than 25 minutes before this process was finished (12 min 41 s for AlexNet, 22 min 16 s for VGG16 and 24 min 44 s for VGG19). As it was the case in the previous approach, the used pretrained networks required only minimal adjustments in the last layers of CNN, before they were redy to use in the presented classification problem. The entire process was not very time consuming, while three resulting models had good generalization properties for drill wear class recognition, both individually, and as the classifier ensemble.

References

[1] K. Jemielniak, T. Urbański, J. Kossakowska, S. Bombiński. Tool condition monitoring based on numerous signal features. Int. J. Adv. Manuf. Technol., vol. 59, pp. 73-81, 2012.

[2] S. S. Panda, A. K. Singh, D. Chakraborty, S. K. Pal. Drill wear monitoring using back propagation neural network. Journal of Materials Processing Technology, vol. 172, pp. 283-290, 2006.

[3] R. J. Kuo, Multi-sensor integration for on-line tool wear estimation through artificial neural networks and fuzzy neural network. Engineering Applications of Artificial Intelligence, vol. 13, pp. 249-261, 2000.

[4] J. Kurek, M. Kruk, S. Osowski, P. Hoser, G. Wieczorek, A. Jegorowa, J. Górski, J. Wilkowski, K. Śmietańska, J. Kossakowska. Developing automatic recognition system of drill wear in standard laminated chipboard drilling process. Bulletin of the Polish Academy of Sciences. Technical Sciences, vol. 64, pp. 633-640, 2016.

[5] J. Kurek, G. Wieczorek, M. Kruk, A. Jegorowa, S. Osowski. Transfer learning in recognition of drill wear using convolutional neural network. 18th International Conference on Computational Problems of Electrical Engineering (CPEE), pp. 1-4. IEEE. September 2017.

[6] J. Kurek, I. Antoniuk, J. Górski, A. Jegorowa, B. Świderski, M. Kruk, G. Wieczorek, J. Pach, A. Orłowski, and J. Aleksiejuk-Gawron. Data augmentation techniques for transfer learning improvement in drill wear classification using convolutional neural network. *Machine Graphics & Vision*, 28(1/4):3–12, 2019. doi:10.22630/MGV.2019.28.1.1.

[7] J. Kurek, B. Świderski, A. Jegorowa, M. Kruk, S. Osowski. Deep learning in assessment of drill condition on the basis of images of drilled holes. Proc. SPIE 10225 Eighth International Conference on Graphic and Image Processing (ICGIP 2016), pp. 102251V, February 8, 2017.

[8] L. Deng, D. Yu. Deep Learning: Methods and Applications. Foundations and Trends in Signal Processing, vol. 7, pp. 3-4, 2014.

[9] Y. Bengio. Learning Deep Architectures for AI. Foundations and Trends in Machine Learning, vol. 2, no. 1, pp. 1-127, 2009.

[10] I. Goodfellow, Y. Bengio, A. Courville. Deep learning. MIT Press, 2016.

[11] J. Schmidhuber. Deep Learning in Neural Networks: An Overview. Neural Networks, vol. 61, pp. 85-117, 2015.

[12] A. Krizhevsky, I. Sutskever, G. Hinton. Image net classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, vol. 25, pp. 1-9, 2012.

[13] O. Russakovsky, J. Deng, H. Su et al. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV), vol. 115, no. 3, pp. 211-252, 2015.

[14] BVLC AlexNet Model. Online: `https://github.com/BVLC/caffe/tree/master/models/bvlc_alexnet`.

[15] Matlab 2017a – User manual, Natick, MA, USA: The MathWorks, Inc, 2017.

[16] ImageNet. Online: `http://www.image-net.org`.

[17] B. Scholkopf, A. Smola. Learning with Kernels, Cambridge: MIT Press, 2002.

[18] M. Kruk, B. Świderski, S. Osowski, J. Kurek, M. Słowińska, I. Walecka. Melanoma recognition using extended set of descriptors and classifiers. Eurasip Journal on Image and Video Processing, vol. 43, pp. 1-10, 2015.

[19] V. N. Vapnik. *Statistical Learning Theory*, New York: Wiley, 1998.

[20] Description of Matlab image transformations. Online: `https://www.mathworks.com/help/deeplearning/examples/image-augmentation-using-image-processing-toolbox.html`.