# Cards recognition on a set of skat deck by the use of convolution neural network

Michalina Owczarek, Roman Kluger, Łukasz Ryziewicz
Institute of Mathematics
Silesian University of Technology
Kaszubska 23, 44-100 Gliwice, Poland
Email: michowc672@student.polsl.pl, romaklu253@student.polsl.pl, ryziewiczlukasz@gmail.com

***Abstract*—In this article we present a model based on convolution neural network for object recognition on a photo. Neural network model was trained on two sets of photos which contain two different patterns of cards in first (larger) set and one card pattern in the second data set. The systems works well to verify images and recognize card on the image.**

## I. Introduction

Information processing makes many possibilities for development of smart systems. New technologies work in various places from industry to computer. We can find applications of methods and models in many places. Neural networks are very often used as detectors for different data types. In technical systems neural networks serve as controllers of geometric features [1], optimization techniques for safe energy [2], simulation methods for complex dynamic systems [3] or control in smart environments [4].

Another important field of application is image processing. Here we can find a wide variety of approaches but applied neural network constructions are much more sophisticated. We can use advanced image processing to search for interesting shapes of bacteria to enable faster recognition in medical laboratories [5]. Other approaches use neural networks to solve problems of identification [6]. However among neural networks used for image processing most efficient results are reported by the use of convolutional neural networks. These structures are devoted to image processing, since first steps are for image filtering and pooling what extracts the set of important information for further processing. A study over many important recent methods by the use of convolutional neural networks was presented in [7]. Recently an important results from convolutional neural networks were also reported in recognition of objects by adjusting these constructions to special conditions of input images [8] or using their deep constructions for complex image reasoning [9].

In this article we present an approach to use neural networks as classification for card game. In our idea convolutional neural network was trained and used to recognize cards from images. The construction we have used is based on classic approach from python language libraries. We have implemented a system which takes an image from smart phone camera and forwards it to server, where a python program works for classification. The results show our idea is interesting and works well.

### A. History of neural networks

Neural networks originate in idea of using methods developed by millenia of evolution. Their development became possible because of deepened research in field of neurobiology and also rapid growth of calculation power of electronic devices.

One of the first scientific description concerning operation of human brain came up in the beginning of 20. century. It was written by Ramn y Cajal who made an assumption that the brain consists of interconnected autonomic parts. Each of these is responsible for different actions. He also wrote about special cells which are processing signals received from senses and also produce signals controlling parts of human body. These cells are called neurons.

About 50 years later another explorer, John Eccles prooved that very important role is played by connections between neurons mentioned above. These are called synapses. Synapses enhance useful signals end decrease another. This part of information processing plays important role in process of learning.

First model of neuron was made by Macculloch and Pitts in 1943. It consists of many input signals activation function and one output signal. In that model a simple Heavisides function was used as an activation function. This model was later developed in 1957 by Frank Rosenblatt and Charles Wightman. They used 8 neurons with 512 connections and build an electromechanical device for image classification. Their device was called Perceptron. In 1960 Bernard Widrow built an Adaptive linear element also called Adaline. It was about ten times faster than the Perceptron. It was also more popular in use, because it could analyze signals from radars and other sensors. It had a large drawback, because it was a linear classifier. It could only recognize linear separable elements. For example a simple xor function is beyond possibilities of these. This problem can be solved by usage of multiple layers of neurons. The fast development of neural networks was unfortunately stopped, because Marvin Minsky and Seymour Papert in 1969 showed in their book, that perceptrons and similar neural networks have limited possibilities of application. Despite this, there wree some examples of interesting neural networks in 1970s. One of them was Brain state in a box built by James Anderson in 1977. It was one of the first examples of neural network used for extracting information from database. In 1982 John Hopfield developed a recurrent neural network for solving problems like traveling salesman problem. Recurrent neural networks are used for making associative memory. The network can learn some patterns and then can associate parts of these patterns to create whole picture. In 1986 David Rumelhart wrote a paper which popularized algorithm of backpropagation of errors. This algorithm allows us to change weights of singular connections between neurons proportionally to error caused by them.

## II. Data set

In this section we would like to present our data used for system implementation.

### A. What the Skat actually is

Skat is a card game for 3 or more players (3 players are active players, others passes hand). To play skat is used skat deck which consist of 32 cards from As to 7th in each suit. There are 4 suits: Acorns, Leaves, Hearts, Bells. Our goal was to create the neural network which could detect card on the photo and recognize which card (or cards) are on the photograph. We were also curious what is the accuracy of the network and how the size of dataset and number of computed steps of network affect on its accuracy.

### B. Appropriate data

We assumed that our Convultional Neural Network will be recognizing objects on photos, so as well as test set, also teaching set consist of photos on which we can see cards and because of program which was used to tagging those cards, laying parallel to edge of the picture.

### C. Sets of data

First of all we prepared two data sets using two different deck patterns.

First data set (the larger one) contains all images of both deck, which is about 300 images and the second set consist only of photos of the first deck counting about 190 pictures.

All cards on every photo was labeled by rectangle box and than all of labels were collected in one file as tensor vector, which neural network could take to compute each weights for all cards.

### D. Data conversion

After completing set of images and labeling it, we collected whole labels in one text file and exported to vector of tensors, which can be read and used by TensorFlow. Data prepared like this could be used now for training convolutional neural network.

**Fig. 1:** Pattern A.

**Fig. 2:** Pattern B.

**Fig. 3:** Other example of photograph from data set.

**Fig. 4:** The same photograph with label boxes on it.

## III. Neural Network Model

We used Faster RCNN with Inception v2 configured for Oxford-IIIT Pets Dataset, which was created to recognize cats and dogs. That was the pretrained model which we have used and retrain on our dataset.

### A. The Oxford-IIIT Pet Dataset

Its a collection of 7,349 images of cats and dogs . Theres 25 different dog breeds and 12 cat breeds. Each breed has about 200 images. Images are divided into following sets: 50 images for training, 50 for validation, 100 for test. Each image is annotated with breed label, a pixel level segmentation marking the body, and a tight bounding box about the label. The segmentation is trimap with regions corresponding to: foreground (the pet body), background, and ambiguous (the pet body boundary and any accessory such as collars).

### B. Evaluation protocol

Three tasks are defined: pet family classification (Cat vs Dog, a two class problem), breed classification given the family (a 12 class problem for cats and a 25 class problem for dogs), and breedand family classification (a 37 class problem).

*1) A model for breed discrimination:*

1) **Shape model** An object is given by a root part connected with springs to eight smaller parts at a finer scale. The appearance of each part is represented by a HOG filter, capturing the local distribution of the images edges; inference (detection) uses dynamic programming to find the best trade-off between matching well each part to the image and not deforming the springs too much.
2) **Appearance model**
   a) **Image layout**-consists of five spatial bins organized as 1x1 and 2x2 grids covering the entire image area. This results in a 20,000 dimensional feature vector.
   b) **Image + head layout**-adds to the *image layout* just described a spatial bin in correspondence of the head bounding box (as detected by the deformable

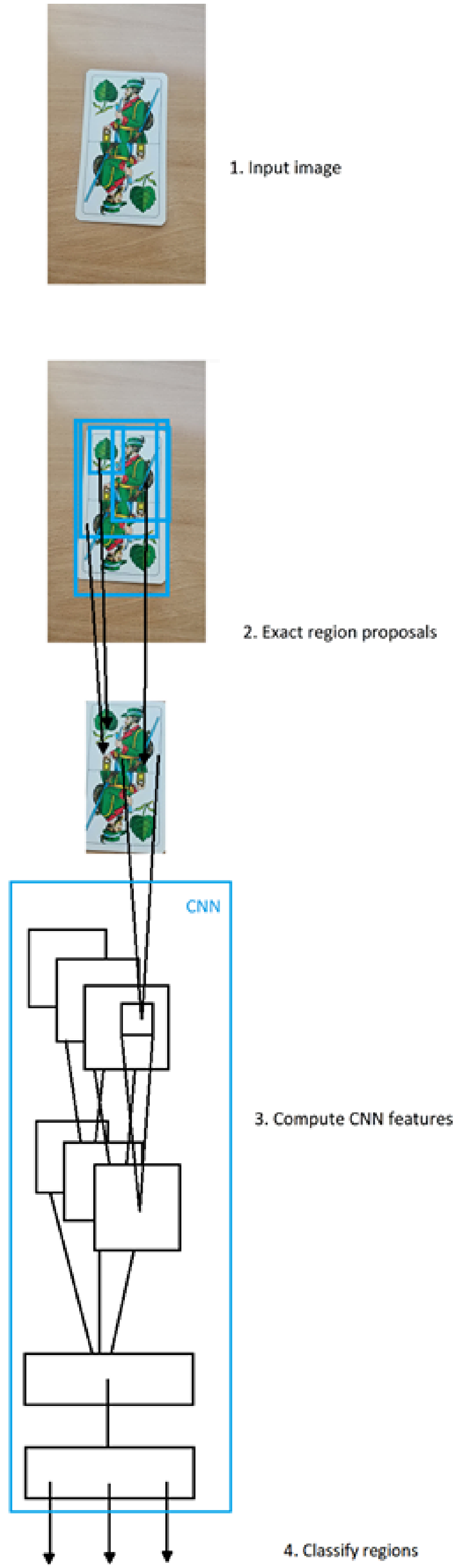


**Fig. 5:** Implemented way of processing card images in our system.

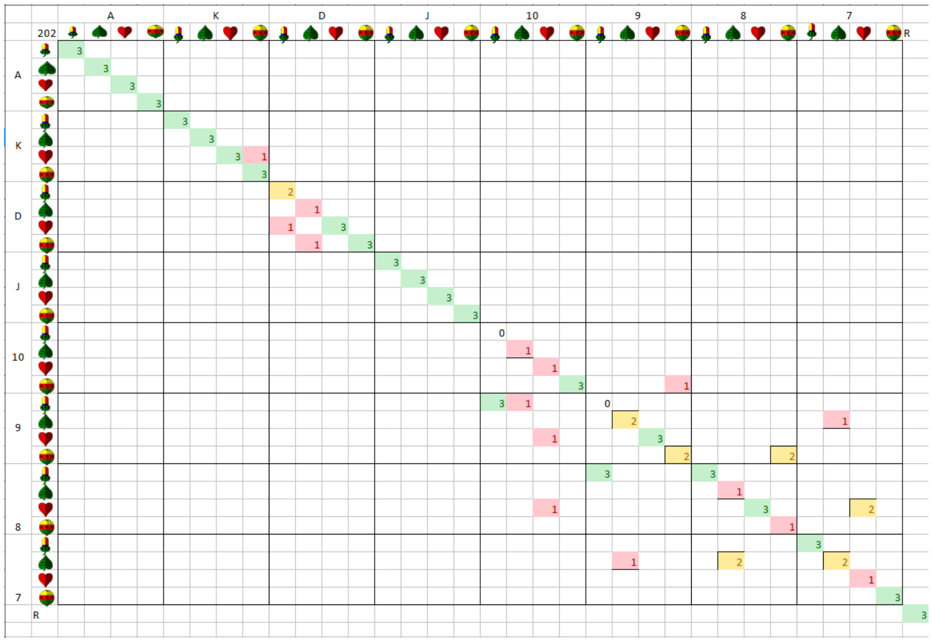

**Fig. 6:** Confusion matrix in experiment 1. Accuracy of convolutional neural network which was learning on two deck patterns by 7000 steps is 78%.

part model of the pet face) as well as one for the complement of this box. These two regions do not contains further spatial subdivisions. Concatenating the histograms for all the spatial bins in this layout results in a 28,000 dimensional feature vector.

c) **Image + head + body layout**- combines the spatial tiles in the image layout with an additional spatial bin in correspondence of the pet head (as for the image+head layout) as well as other spatial bins computed on the foreground object region and its complement (as described next). The foreground region is obtained either from the automatic segmentation of the pet body of from the ground-truth segmentation to obtain a best-case baseline. The foreground region is subdivided into five spatial bins, similar to the image layout. An additional bin obtained from the foreground region with the head region removed and no further spatial subdivisions is also used. Concatenating the histogram for all the spatial bins in this layout results in a 48,000 dimensional feature vector.

The foreground (pet) and background regions needed for computing the appearance descriptors are obtained automatically using the grab-cut segmentation technique.

### *C. Experiments*

The models are evaluated first on the task of discriminating the family of the pet, then on the one of the discriminating their breed given the family, and finally discriminating both the family and the breded. For the third task, both hierarchical classification (i.e, determining the family and the breed simultaneously) are evaluated. Training uses the the Oxford-IIIT Pet train and validation data and testing uses the Oxford-IIIT Pet test data.

*1) Pet family discrimination:*

1) **Shape only**- The maximum response of the cat face detector on an image is used as an image-level score for the cat class. The same is done to obtain a score for the dog class. Then a linear SVM is learned to discriminate between cats and dogs based on these two scores. The classification accuracy of this model on the Oxford-IIIT Pet test data is 94.21%.
2) **Appearance only**- Spatial histograms of visual words are used in a non-linear SVM to discriminate between cats and dogs. The accuracy depends on the type of spatial histograms consider, which in turn depends on the layout of the spatial bins. On the Oxford-IIIT Pet test data, the image layout obtains an accuracy of 82,56%; adding head information using image+head layout yields an accuracy of 85.06%. Using image+head+body layout improves accuracy by further 2.7% to 87.78%. An improvement of 1% was observed when the ground-truth segmentation were used i place of the segmentation estimated by grab-cut. The progression indicates the more accurate the localization of the pet body, the better is the classification accuracy.
3) **Shape and appearance**- The appearance and shape information are combined by summing the $exp - \chi^2$ kernel for the appearance part with linear kernel on the cat scores and a linear kernel on the dog scores. The combination boosts the performance by an additional 7% over that of using appearance alone, yielding approximately 95.37% accuracy, with all the variants of

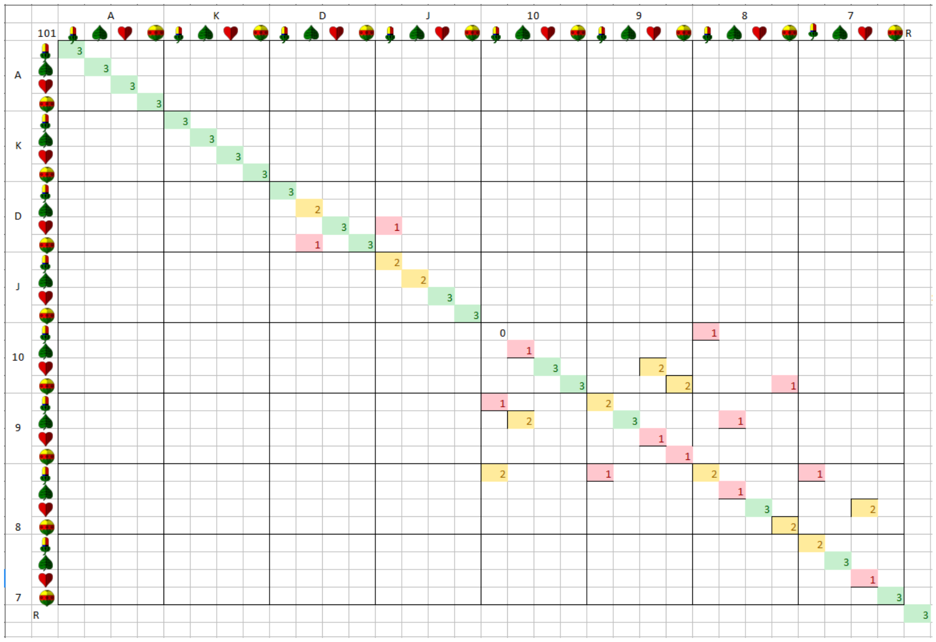


**Fig. 7:** Confusion matrix in experiment 2. For one deck pattern and teaching for 7000 steps accuracy is 80%.

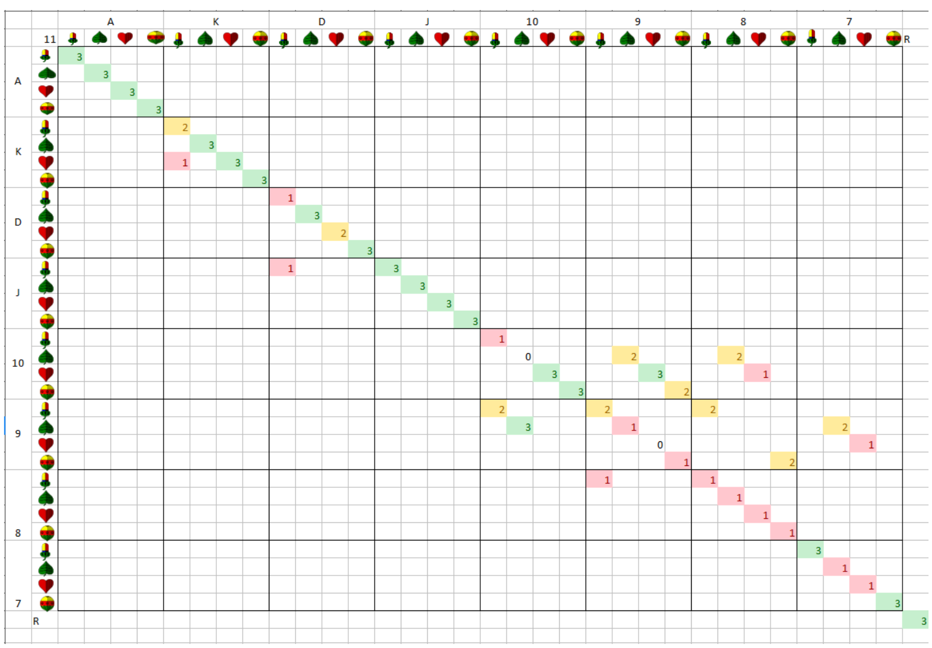


**Fig. 8:** Confusion matrix 3. Using two deck patterns and teaching 9000 steps neural network is on 71% level.

the appearance model performing similarly

*2) Breed discrimination:* This section evaluates the model on the task of discriminating the different breeds of cats and dogs given their family.This is done by learning a multi-class SVM by using the !-Vs-rest decomposition (this mean learning 12 binary classifiers for cats and 25 for dogs). The relative performance of the different models is similar to that observed for pet family classification. The best breed classification accuracies for cats and dogs are 63.48% and 55.68% respectively, which improve to 66.07% and 59.18% when the ground truth segmentations are used.

*3) Family and breed discrimination:* This section investigates classifying both the family and the breed. Two approaches are explored: hierarchical classification, in which the family is decided first and then the breed is decided and flat classification, in which a 37-class SVM is learned directly, using the same method discussed. The relative performance of the different models is similar to that observed. Flat classification is better that hierarchical, but the latter requires less work at test time, due to the fact that fewer SVM classifiers

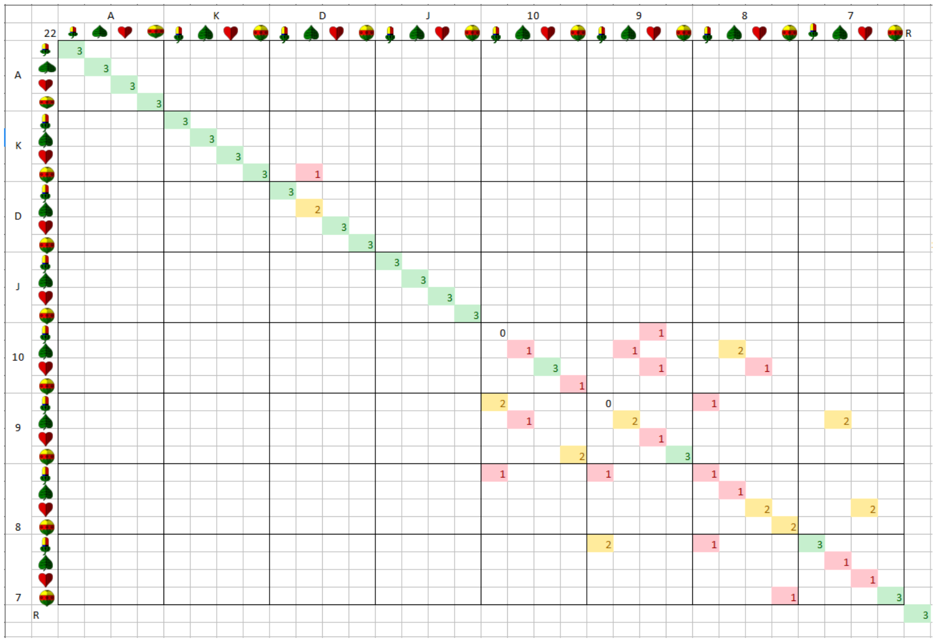


**Fig. 9:** Confusion matrix 4. Network which was learning one deck pattern and doing it for 9000 steps is recognizing proper cards for 76%.

need to be evaluated. Form example, using the appearance model with the image, head, image-head layouts for 37 class classification yields an accuracy of 51.23%, adding the shape information hierarchically improves the accuracy to 52.78%, and using shape and appearance together in flat classification approach achieves an accuracy 54.03%.

## IV. Results

We have been training our network for 7000 and 9000 steps and results as we can see on confusion matrices are as following in figure 6 - figure 9. It is worth to mention that all tests were made on pattern A deck.

### A. Conclusions

As we can see, there are small but important differences, which tell us that better results can be reached by increasing teaching time or by reducing the variety of objects in dataset.

In our future research we want to investigate other methods of image processing for pattern recognition. We are interested in other architectures of convolutional neural networks and other python libraries developed for classification.

## References

[1] G. L. Sciuto, G. Capizzi, C. Salvatore, and R. Shikler, "Geometric shape optimization of organic solar cells for efficiency enhancement by neural networks," in *Advances on Mechanics, Design Engineering and Manufacturing*. Springer, 2017, pp. 789–796.

[2] G. Capizzi, G. L. Sciuto, C. Napoli, and E. Tramontana, "Advanced and adaptive dispatch for smart grids by means of predictive models," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6684–6691, 2017.

[3] U. R. Acharya, H. Fujita, S. L. Oh, U. Raghavendra, J. H. Tan, M. Adam, A. Gertych, and Y. Hagiwara, "Automated identification of shockable and non-shockable life-threatening ventricular arrhythmias using convolutional neural network," *Future Generation Computer Systems*, vol. 79, pp. 952–959, 2018.

[4] M. Woźniak and D. Połap, "Intelligent home systems for ubiquitous user support by using neural networks and rule based approach," *IEEE Transactions on Industrial Informatics*, 2019.

[5] M. Woźniak, D. Połap, L. Kośmider, and T. Cłapa, "Automated fluorescence microscopy image analysis of pseudomonas aeruginosa bacteria in alive and dead stadium," *Engineering Applications of Artificial Intelligence*, vol. 67, pp. 100–110, 2018.

[6] D. Połap, M. Woźniak, R. Damaševičius, and R. Maskeliūnas, "Bio-inspired voice evaluation mechanism," *Applied Soft Computing*, vol. 80, pp. 342–357, 2019.

[7] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.

[8] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, vol. 106, pp. 249–259, 2018.

[9] S. R. Kheradpisheh, M. Ganjtabesh, S. J. Thorpe, and T. Masquelier, "Stdp-based spiking deep convolutional neural networks for object recognition," *Neural Networks*, vol. 99, pp. 56–67, 2018.