Department of education of Chisinau

N.M. Spataru Lyceum

**Skin Cancer Early Diagnosis using Transfer Learning.**

Author:

Melnic Roman

Instructor:

Golubev Svetlana

Chisinau, 2019

1. INTRODUCTION

Skin cancer is the most common disease in the world. Melanoma is one of the deadliest and dangerous forms of skin cancer[1]. In 2018 there was 287,000 new causes of melanoma[2]. On late stages, when process of metastasis begins, survival rate is only 10-20%[3]. Though, if it is treated on early stages, survival rate is more than 90%. That’s why early diagnosis of melanoma is so important. For melanoma diagnosis it is necessary to take a physical exam by a dermatologist. If the doctor will think that it might be a malignant lesion, a biopsy of it will be needed to find cancer cells. However very often on early stages melanoma lesions are similar to normal moles and don’t look suspicious to people, therefore they don’t go to the doctor for treatment.

Nowadays deep learning is growing fast and is used in many areas of human activities, including medicine. It is already applied for medical image analysis and disease diagnosis.[4] Now a lot of machine learning models that can classify benign and malignant moles with high accuracy exist, but common people that don’t have much knowledge in machine learning or programming can’t easily use them because of the complexity that is required to run them.

The scope of my project is to create a deep learning model using a transfer learning technique that could classify melanoma with high accuracy and then to embed it in a mobile application that could be easily used by common people for checking their moles, therefore it will improve the chance of early diagnosis of skin cancer.

1. TASKS

To successfully achieve the scope of the project the following tasks are required:

-To learn about machine learning algorithms.

-To choose the right model for future retraining.

-To create an architecture of neural network.

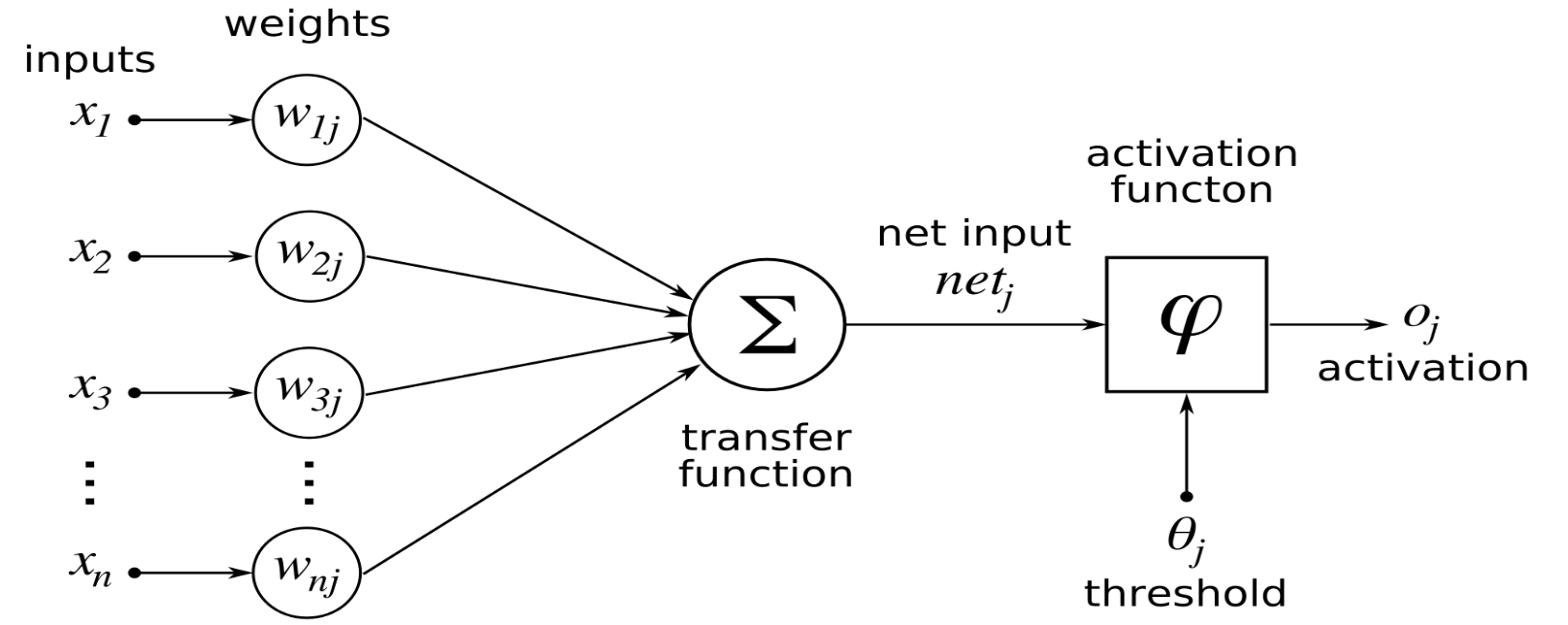
-To create a dataset of benign and malignant images.

-To train the classifier.

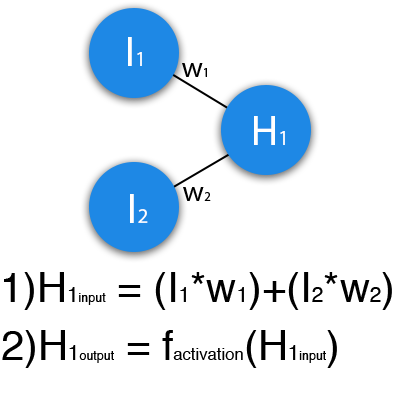
-To create and to test the mobile application using a trained classifier.

1. ANN

Artificial neural networks(ANN) is a popular method of machine learning, inspired by structure of human brain. They consist of mathematical representation of neurons and links between them that have weights. Image below illustrates it.



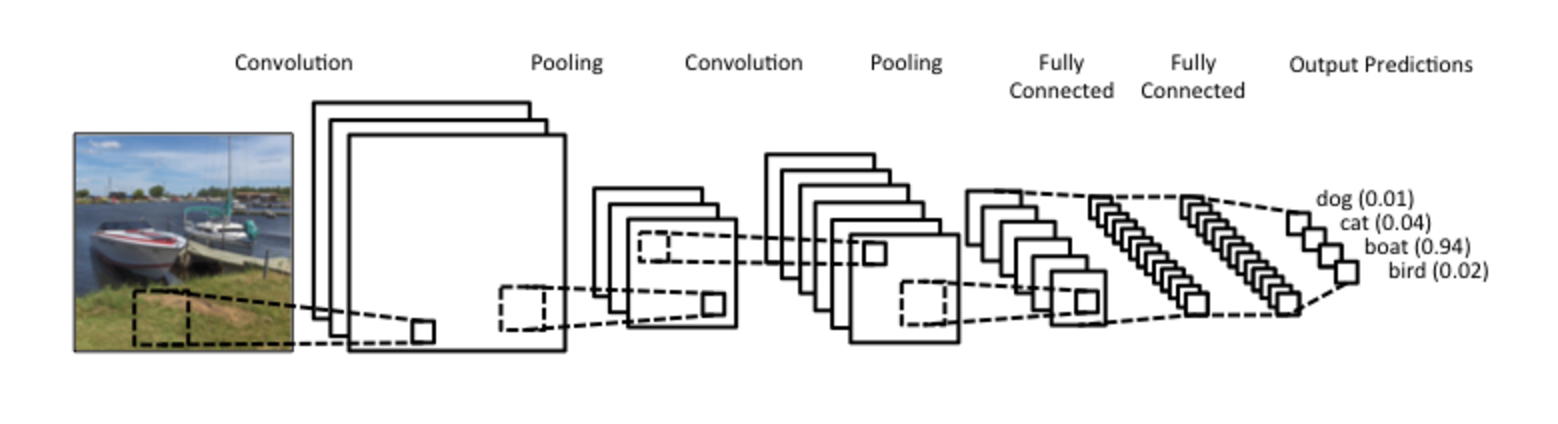
On the input neuron gets some value that is multiplied by weight of the connection, than all inputs are added up and processed by activation function. Output value is given to other neurons as input.

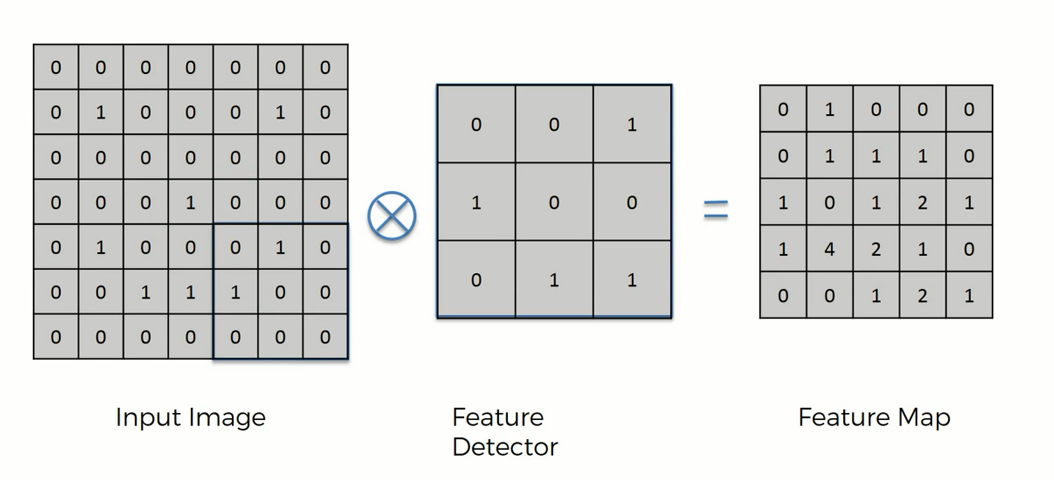
In ANN neurons are combined in layers. There are 3 types of layers : input, hidden and output. Input layer gets raw data that is processed by hidden layers and than is given to output layers. Network can contain only 1 input and output layer, but many hidden layers. ANNs can analyze and even remember data, so they are widely used in classification, regression and recognition problems.

1. CNN

Convolutional neural network(CNN) Is a specific type of ANN designed for image processing. Before data goes to normal neuron layers, it is processed by convolutional and pooling layers where some features of image are extracted.

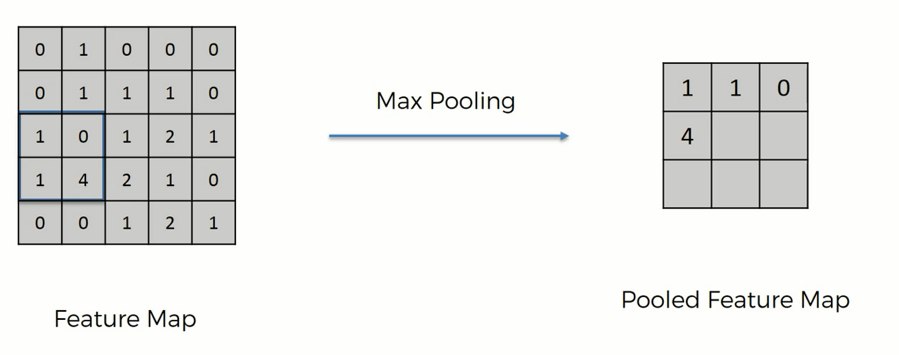


Process of convolution consists of applying some matrix filters, called feature detectors, on image and creating feature maps. It is done for extracting from image some properties or characteristic which can be analyzed by neural network.

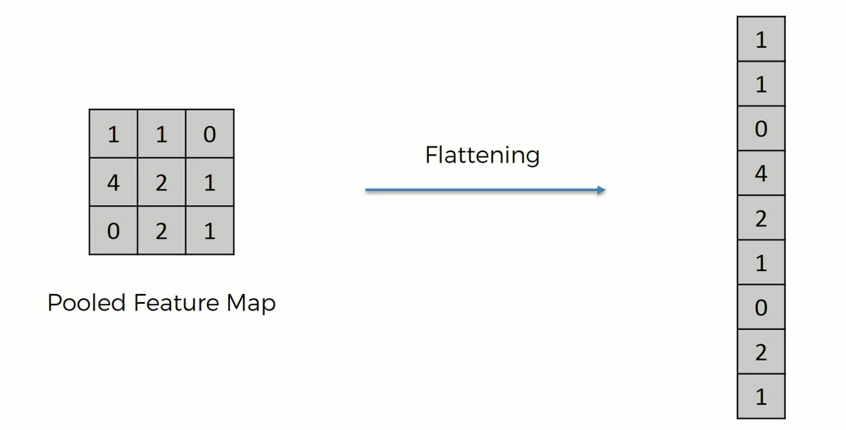


Filter, or kernel, goes through whole image with some stride. Values from filter are multiplied by values from the image, then the sums of values form a feature map.

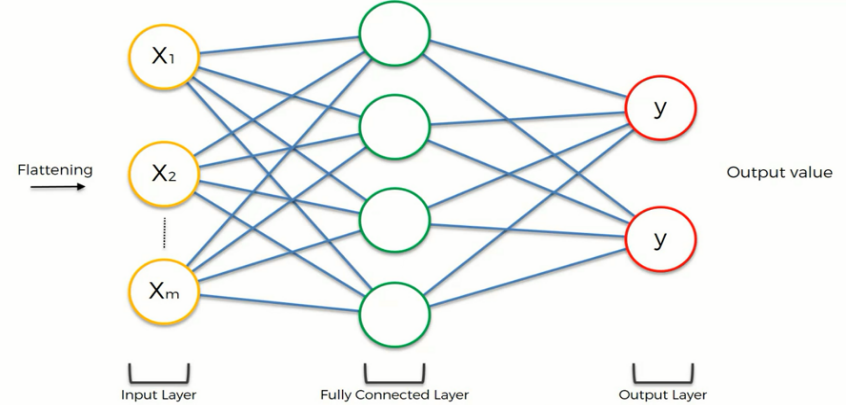
On every convolutional layer many maps are created, that usually are then processed by pooling layers. Polling operation reduces dimensionality of image while keeping all the features and making network more resistant to shifts.



In pooling layers filters are applied in same way, usually with 2x2 dimensions. They make with values of image some operations and create reduced feature map. On image above is illustrated max pooling filter that takes maximum of values in its working area. After that, all feature maps are flattened to one big array.



It is used as input for normal neural network.

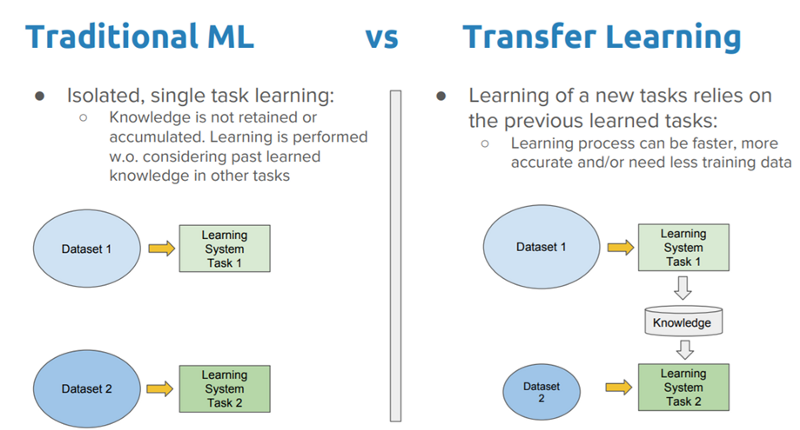


Thereby convolutional layers extract features from image and normal layers analyze them.

1. TRANSFER LEARNING

Humans have an inherent ability to transfer knowledge across tasks. What we acquire as knowledge while learning about one task, we utilize in the same way to solve related tasks. The more related the tasks, the easier it is for us to transfer, or cross-utilize our knowledge. For example, when you know how to play classic piano, It is easier to learn how to play jazz piano, or if you know Italian, it is easier to learn Spanish. In that scenarios, we don't learn everything from scratch, we transfer and leverage our knowledge from what we have learnt In the past[5].

Machine learning models are more likely to train from scratch for resolving specific and focused tasks. However, many studies proved that transferring of knowledge can be achieved with machine learning[6]. The idea of transfer learning consists in retraining a model, that was previously used to solve one problem to solve another, similar problem. In that way we can achieve faster learning speed and can reduce the necessity of big amounts of training data.

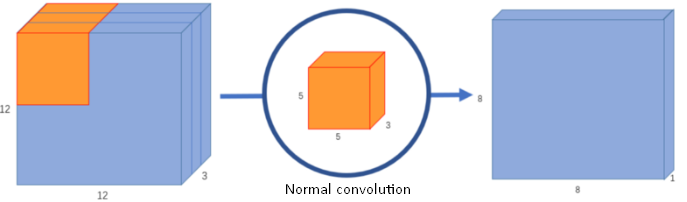


There are two basic methods of transfer learning : freezing the base model weights or fine-tune them. In the first case the weights of the base model don’t change while training and are used as a feature extractor, in the other case we train them too.

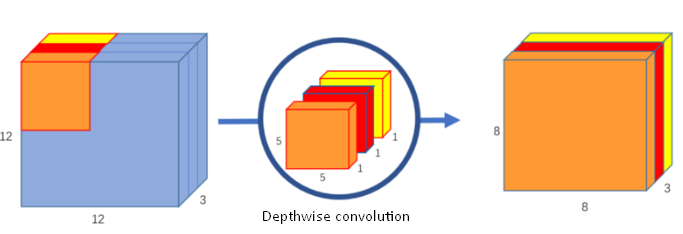
1. MOBILENETv2

There are many pretrained models for image classification already created. One of them is called MobileNetV2, which architecture consists of expansion convolution blocks[7].

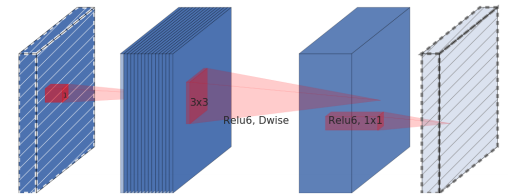
Initially, raw data goes through expansion convolutional layer, where on image is applied many pointwise 1x1 filters, therefore generating many feature maps. Next goes depthwise convolutional layer. In normal convolutional layers image channels are processed together.



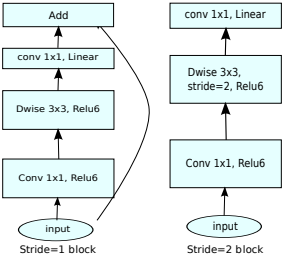
However in depthwise layers, channels are processed separately with different filters.



After that dimensionality of image is reduced by applying 1x1 pointwise convolution, but with few filters and linear activation function.

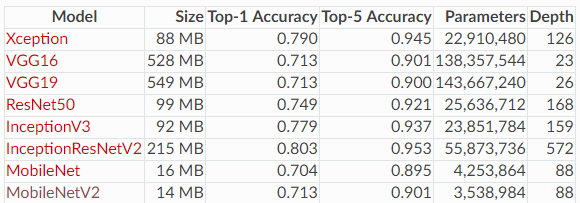


MobileNetV2 uses 2 types of expansion blocks : with stride of 1 and with stride of 2. Second one is used for dimensionality reduction, because model doesn’t has pooling layers.



Advantage of using depthwise layers instead normal ones are in much less amount of computational operations. For example, we have 12x12x3 RGB image and we want to create 256 feature maps using 5x5x3 filters. If they will move with stride of 1, then 8x8 steps will be needed to compute one map, and this will be 256x5x5x3x8x8=1,228,800 operations of multiplication. Using depthwise layers, every channel will be processed separately by 5x5x1 filter 8x8 times, this is 5x5x1x8x8=4,800 operations. As the result we will have 8x8x3 image, and to create from it 256 feature maps, we will need to use pointwise 1x1x3 filter 8x8 times, that’s 256x8x8x1x1x3=49,152 multiplications. Adding them up, that’s 4,800+49,152=53,952 multiplications, which is a lot less than 1,228,800[8].

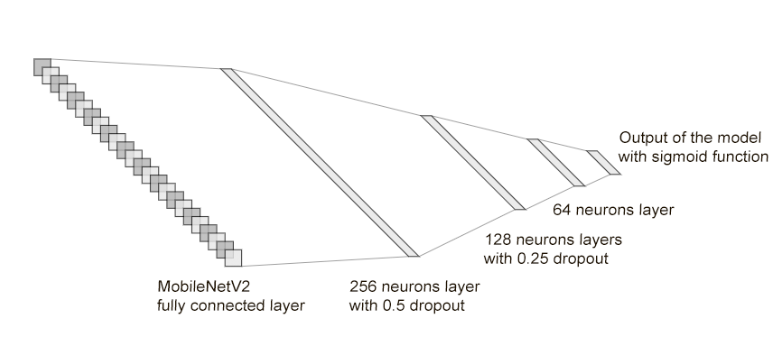
MobileNetV2 was trained on largest labeled image dataset Imagenet. Image below shows its results in comparison to other models [9].



It has only 3.5 millions parameters and size of only 14MB! And accuracy is pretty high. This allows to apply it effectively and quickly on mobile devices, and that’s why It is used in this project.

1. MODEL STRUCTURE

MobileNetV2 was chosen as a base model, her weights, trained on Imagenet were frozen, and after them 3 hidden layers were added to train a classifier. The image below shows the scheme of the model.



In the first two hidden layers the dropout technique was applied, when in every training iteration some percentage of neurons don’t change their weights, it helps to prevent overfitting, when model simply remembers training data, but not actually learns.

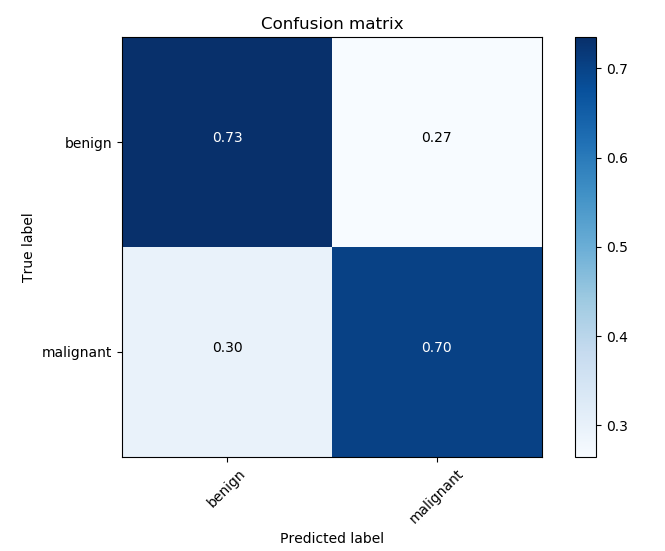
We only have two classes, malignant and benign images, therefore on the output layer the sigmoid activation function was used, it generates the probability of melanoma presence on the image. For creation and training the model, Python machine learning framework Keras was used.

1. TRAINING

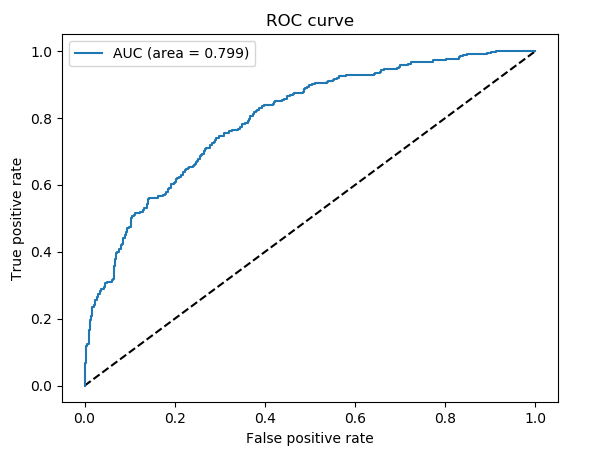
For training data two medicine datasets from the ISIC[10] archive were used, HAM10000 and MSK-2. Using data augmentation, the training dataset was created, containing 2,000 malignant and 2,000 benign images, and the test dataset with 400 images in each category. Images were cropped to 128x128, normalized and given to the model input layer. The model was trained with 200 epochs for about 9 hours on the GTX860M graphic card.

1. METRICS

In the end of training the model showed 80% accuracy on training set and 75% accuracy on test set, which is very good, considering that accuracy of certified dermatologists in classification of melanoma by physical exam is around 75-85%[11]. On images below some metrics of the model are shown.



The first image illustrates the confusion matrix. The first row contains the values of the benign images. In the first column is the percentage of benign images that were predicted correctly and in second one that were predicted incorrectly. The second one row contains values of the malignant images. In the first column is the percentage of the malignant images that were predicted incorrectly and in the second one that were predicted correctly.



Moreover, a ROC(Receiver operating characteristics) curve was created and the AUC(Area under curve) was calculated. The model is giving us a percentage of malignancy of an image, therefore to calculate its accuracy we need to put a threshold, that will separate probabilities that are malignant and probabilities that are benign. For image accuracy computation a threshold of 50% was chosen, but it doesn’t fully show how well the model is performing, and this is where the ROC curve will help us. It illustrates the false positive values (which are benign, but the model predicted them as malignant) on the X axis and the true positive values(malignant images, that the model predicted correctly) on the Y axis, which are computed using different thresholds. The black line in the middle shows the performance of random guessing, the blue line shows the performance of our model. The bigger the area under this line, the better the accuracy of our model. In our case it is almost 80%, that is very good for such complicated problem, but there is always a room for improvement.

10. MOBILE DEPLOYMENT

After the successful training of the model, we need to create a mobile app, that will get an image of mole from a camera, and give it to the model and show the results. For these tasks Xamarin Forms framework was selected, because it allows to create a cross-platform mobile application with shared code base for Android and IOS. Since for the creation of the neural network we used MobileNetV2, it can easily run offline on mobile processors. The weights of the network have the size of about 10MB and predictions take less than a second. On the image below is illustrated screenshot of Android application.



The application was only tested on Android, because of the lack of an IOS device, but It can be easily compiled and tested on IOS too.

11. CONCLUSION AND FUTURE PLANS

In the end a lightweight and user friendly application for checking suspicious moles was created. It isn’t 100% accurate, but it can help people to find melanoma on early stages and to begin treatment.

12. BIBLIOGRAPHY

1. Chante Karimkhani, Adèle C Green, Tamar Nijsten, MA Weinstock, Robert P Dellavalle, Morteza Naghavi, and C Fitzmaurice. The global burden of melanoma: results from the global burden of disease study 2015. British Journal of Dermatology, 177(1):134–140, 2017.
2. Bray F, Ferlay J, Soerjomataram I, Siegel RL, Torre LA, Jemal A. Global Cancer Statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries.
3. Survival Rates for Melanoma Skin Cancer, American Cancer Society, 2016.
4. Deep Learning for Medical Image Processing: Overview, Challenges and Future, Muhammad Imran Razzak, Saeeda Naz and Ahmad Zaib.
5. A Comprehensive Hands-on Guide to Transfer Learning with Real-World Applications in Deep Learning, Dipanjan (DJ) Sarkar, 2018.
6. Use Transfer Learning For Efficient Deep Learning Training On Intel® Xeon® Processors, Beenish Zia, Ramesh Illikkal, Bob Rogers, 2018.
7. MobileNetV2: Inverted Residuals and Linear Bottlenecks, Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen, 2018.
8. A Basic Introduction to Separable Convolutions, Chi-Feng Wang, 2018.
9. Keras Framework Documention, 2018(<https://keras.io/applications/>).
10. The International Skin Imaging Collaboration, 2018.
11. Skin Lesion Analysis Toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging (ISBI), Hosted by the International Skin Imaging Collaboration (ISIC), Noel C. F. Codella, David Gutman, M. Emre Celebi, Brian Helba, Michael A. Marchetti, Stephen W. Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, Allan Halpern, 2017.

