**Real-time rock, paper, scissors recognition project and winning move suggestions**

**Team member:** Makhan Azat IT-2206

**Introduction:** Before we begin the article, it is important to recognize the importance of the project, because creating a working game with winning recommendations is not the way this project is developing. The main idea of the project is to interact with the machine using knowledge taken from the advanced programming course and secondary sources, which are implemented using machine learning and deep learning. In my project, it is done by training a model using the dataset.

**Idea description:** The main idea of the project is to create a machine learning system that analyzes human gestures from an image in real time and, taking into account winning combinations, recommends a winning move to the user.

**Literature review:** As a similar solution to this project, the tenso.rs website was found, which has a similar way of working. However, there the game is played against the computer, where any combination can occur, for example, a win, loss, or draw. Moreover, the demo version is based on the Tensor Force library with GPU acceleration for deep learning in the browser. It is fast enough to classify the video from the webcam on the client side in real-time.

Reference: <https://tenso.rs/demos/rock-paper-scissors/>  
  
Also, after another study, several sites were discovered that implemented the principle of this project on the same TensorFlow, and the information that is published on these sites will be useful for further development of the project and creating a model for the project.

Reference: <https://medium.com/@sdwiulfah/rock-paper-scissors-image-classification-with-keras-tensorflow-1c29ba0fe14d>

<https://medium.com/geekculture/rock-paper-scissors-image-classification-using-cnn-eefe4569b415>

**Current work:**

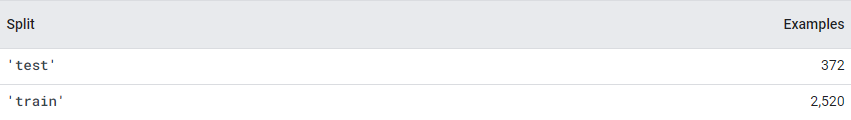
After defining the project topic, the following steps must be performed:

* Data preparation: Data validation for existing files and images for convenient and secure use
* Researching existing Solutions: Find out which existing projects are available and how goals have been achieved
* Building a model: creating a training model for recognizing various gestures (rock, scissors, paper). Improving the model for more correct information output
* Product Introduction: Creating a web application for users with a user-friendly interface.
* Testing and validation: Thorough testing is necessary to ensure the correctness and reliability of recommendation systems.
* Reporting and documentation: Thorough reporting is necessary to properly present the objectives and conclusions of the project.

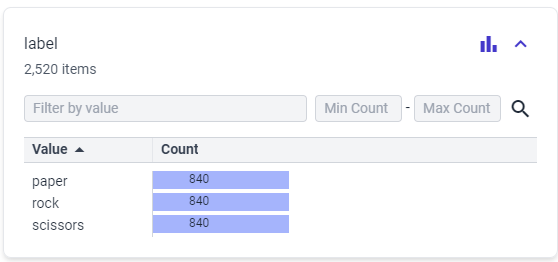
**Data and methods:**

The official TensorFlow datasets were selected as the data set for the project. There is a dataset in the catalog called rock\_paper\_scissors, which is a convenient dataset because you can directly connect the dataset to the Google colab project. It also contains about 3 thousand photos that will give an accurate model after training. **Link:** <https://www.tensorflow.org/datasets/catalog/rock_paper_scissors?hl=en>

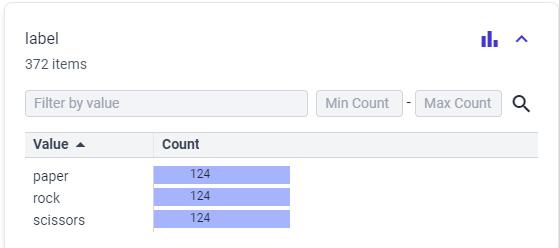
**Information about the data:**



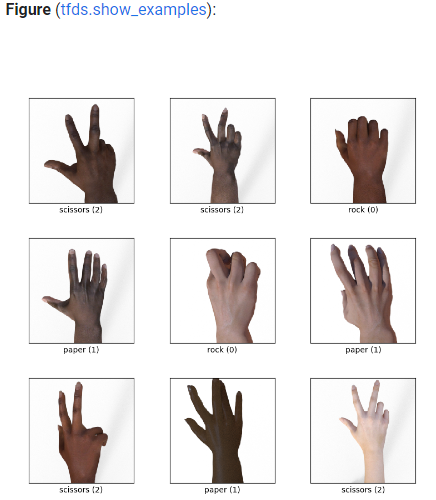
*Figure 1.1 “ Train and test folders with containing image numbers ”*



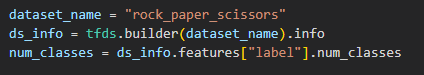
*Figure 1.2 “ Training folder information”*



*Figure 1.3 “ Test folder information”*



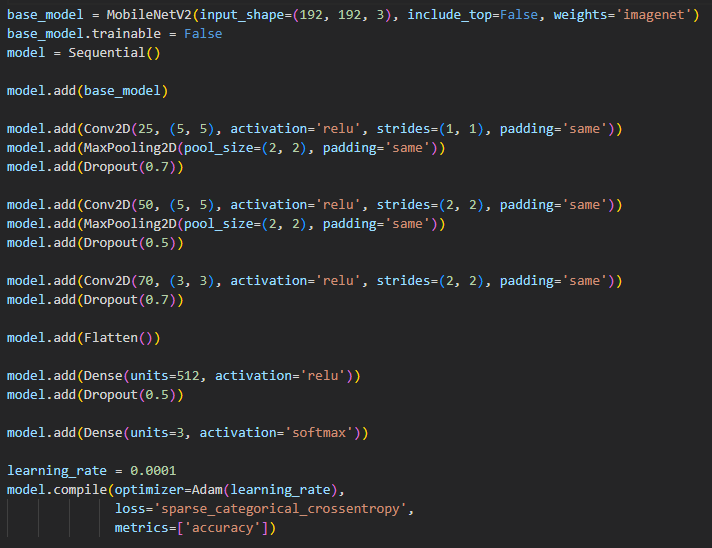
*Figure 1.4 “ Data set examples ”*



*Figure 1.5 “ Data set connection to the google colab ”*

Here I connect the dataset to the google colab for further using.

**Description of the ML/DL models:**



*Figure 1.6 “Code of the model building”*

Here I am creating an instance of Mobile Net V2, a pre-trained convolutional neural network model for image classification. Then I create a sequential model, which is a linear set of layers. The Mobile Net V2 model is added as the first layer of the sequential model. Additional convolutional layers with maximum union and compartments have been added to further extract objects and reduce dimensionality. These layers help to study hierarchical objects based on input data. The model is compiled using the Adam optimizer with a given learning rate, sparse categorical cross entropy as a loss function (suitable for integer labels), and accuracy as an evaluation indicator. This prepares the model for training.

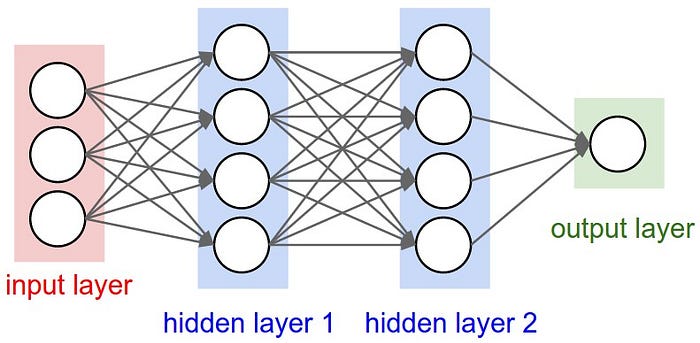


*Figure 1.7 “Code of the model building”*

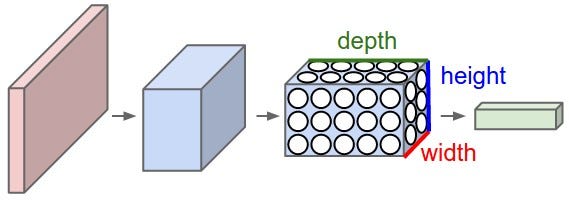
The provided code defines the preprocessing function for the image dataset. The function returns the final preprocessed image and its original label.

The augment\_data of these random transformations is to create variations of the input images, which helps to improve the neural network's ability to generalize during training.

At the end, we adjust the model according to the specified parameters



*Figure 1.8 “A regular Neural Network representation”*



*Figure 1.9 “CNN representation”*

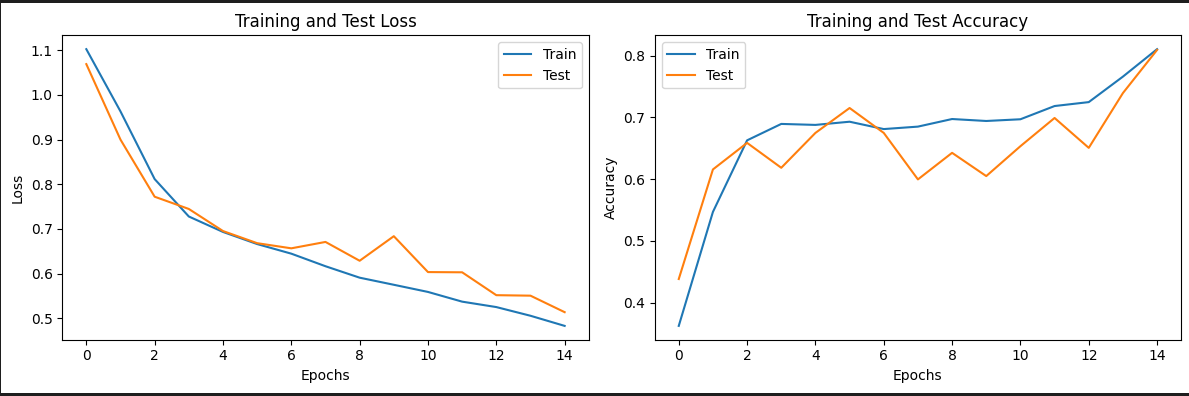
Convolutional neural network (CNN for short) is one of the most popular image classification methods.

CNNs are very similar to the usual neural networks from the previous chapter: they consist of neurons that have trainable weights and offsets. Each neuron receives some input data, performs a scalar product and, if necessary, follows it with non-linearity. The entire network still expresses a single differentiable evaluation function: from the pixels of the raw image at one end to the class ratings at the other. And they still have a loss function (e.g. SVM/Softmax) on the last (fully connected) layer, and all the tips/tricks that we have developed for training conventional neural networks are still applicable.

The difference is that CNN architectures explicitly assume that the input data are images, which allows us to encode certain properties in the architecture. This then makes the forwarding function more efficient to implement and significantly reduces the number of parameters in the network.

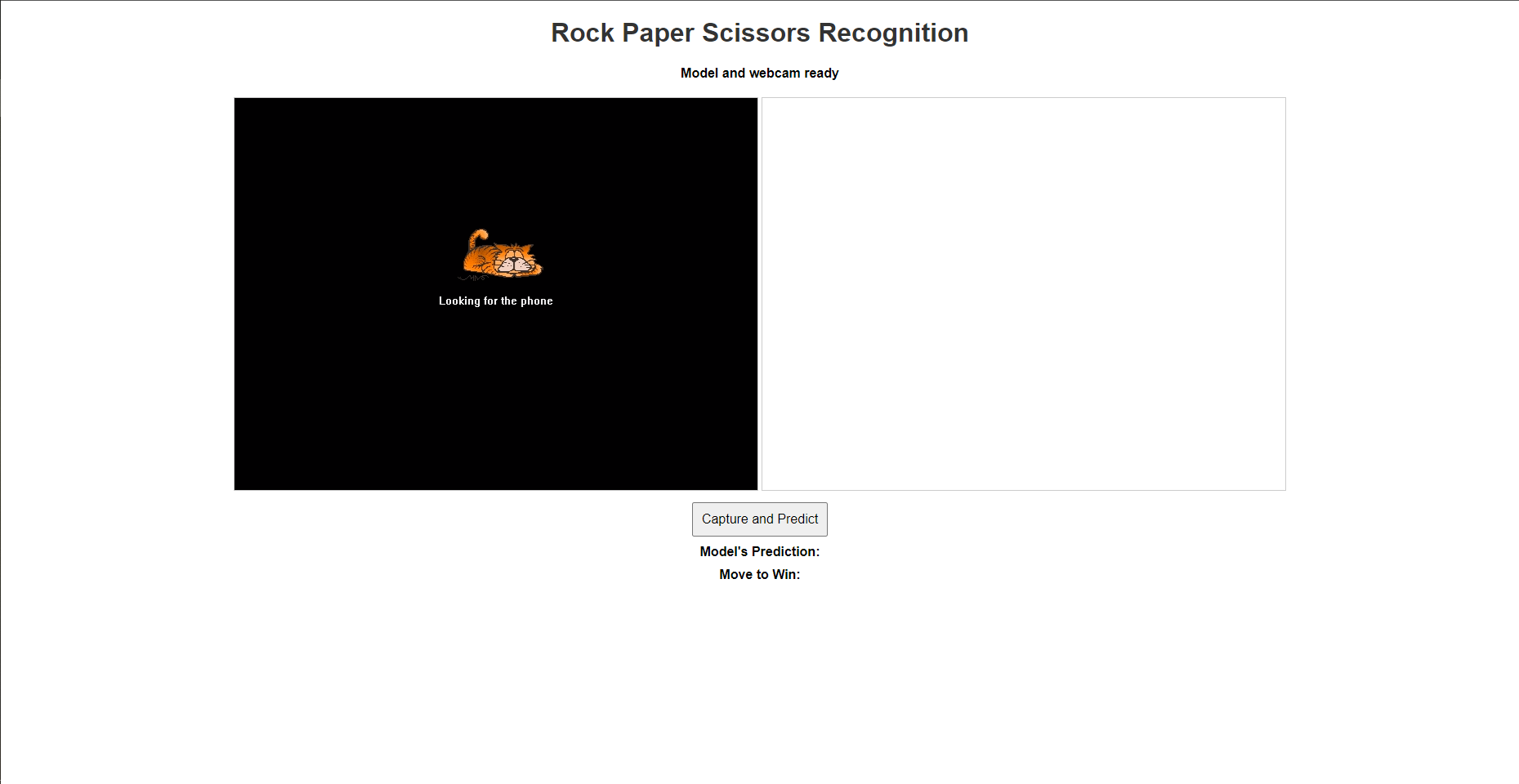
**Results with tables, pictures and interesting numbers:**

In the Results section, let's start analyzing the model as shown in Figure 1.10. Losses decrease and accuracy increases, which means that the model training is completed well, as well as training and testing results in the near range, which means no retraining.

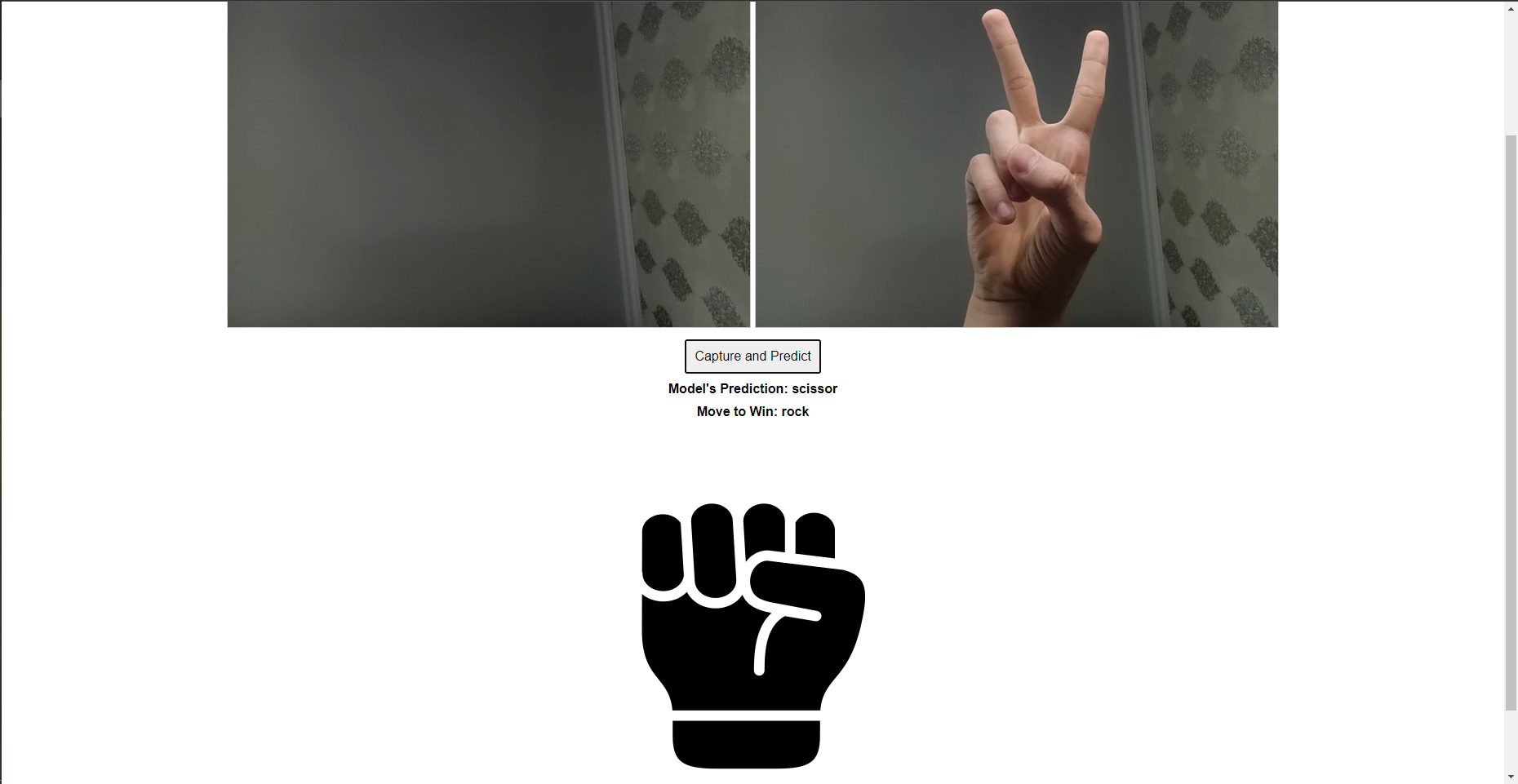


*Figure1.10 “Training and test loss/epoch graph”*

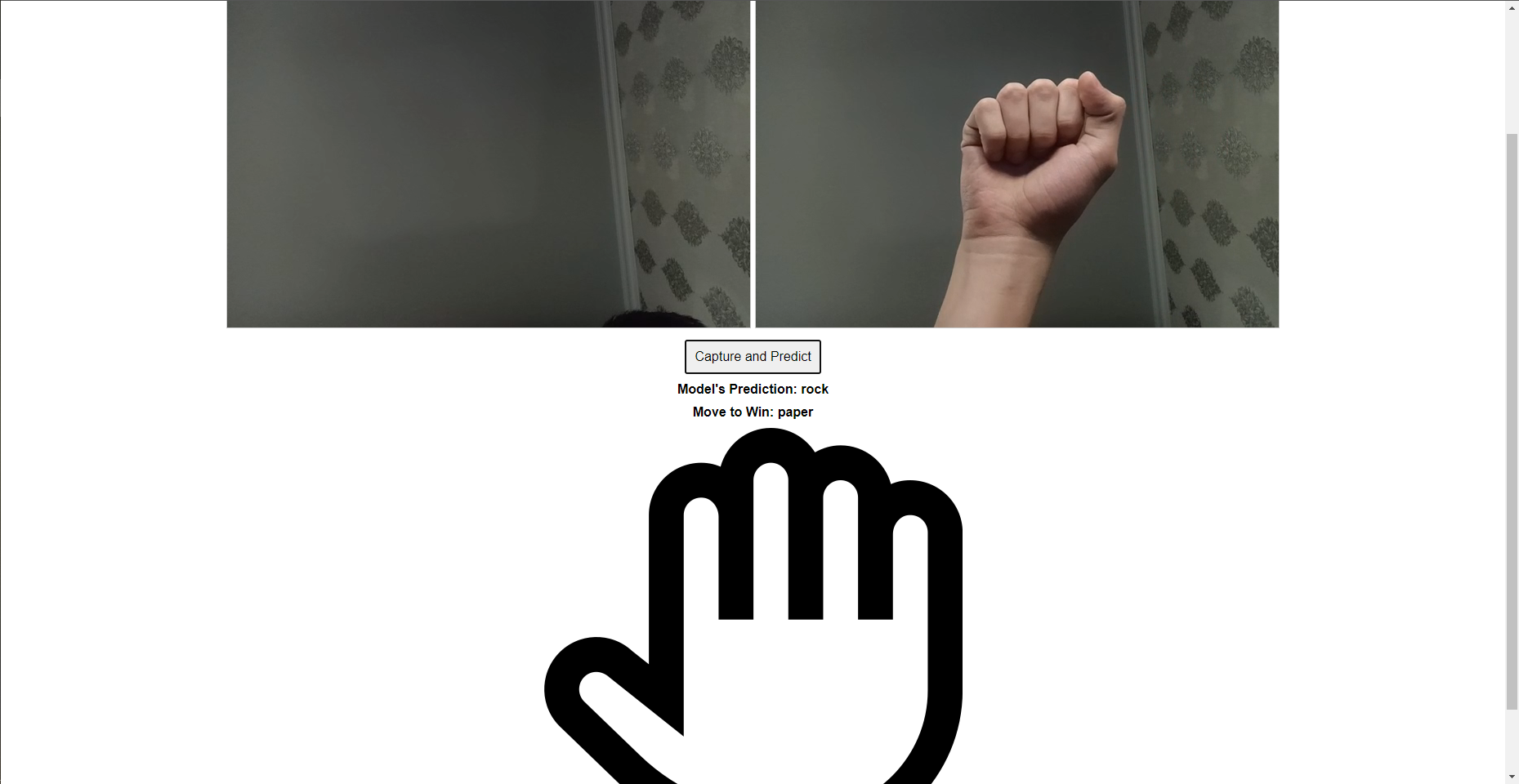
Here is the web application form, in which we have two blocks left: one for the webcam, and the second for the image after capture. After capturing the gesture, it will predict the gesture and then output the winning move with the photo



*Figure 1.11 “web application”*



*Figure 1.12 “web application testing”*



*Figure 1.13 “web application testing”*

The tests were carried out in Figures 1.12 and 1.13, and as you can see in these figures, everything works according to the principle that was laid down at the beginning of this project, which means that we have completed the task.

**Discussion**

As I said earlier, the main idea of the project is to interact with a machine using knowledge taken from an advanced programming course and secondary sources that are implemented using machine learning and deep learning. In my project, this is done by training a model using a dataset. As you can see in the results of the project, everything works according to the principle that was laid down at the beginning. But despite this, it is possible to further improve this project and such ways will be described in the next paragraph.

**Further development**

Since this is a small project for our course, it can be improved for further use by real people, which will avoid a small project to make a product that will be useful for people. Here are a few types of project development:

* Optimize the model with various architectures and hyperparameters.
* Develop a user-friendly interface with HTML/CSS and JavaScript.
* Implement live webcam frame processing for real-time gesture recognition.
* Fine-tune the model for lighting and orientation variations.
* Develop a logic to recommend winning moves based on recognized gestures.

**In conclusion**In conclusion, the rock, paper, scissors recognition project serves as a successful demonstration of the application of machine learning and deep learning principles to create an interactive and attractive system. The described improvements represent a roadmap for converting this course project into a more advanced and user-friendly product for wider use. Overall, the project demonstrates the potential of integrating advanced programming skills with machine learning to create practical and interactive solutions.