****

Faculty of Computers and Artificial Intelligence

Computer Science Department

2021/2022

**CS 396 Selected Topics in CS-2**

**Research Project**

Report Submitted for Fulfillment of the Requirements and ILO’s for Selected Topics in CS-2 course for Fall 2021

Team ID No. 49

|  |  |  |  |
| --- | --- | --- | --- |
|  | ID | Name | Grade |
|  | 20180185 | حازم حسن محمود |  |
|  | 20180378 | عمر فؤاد حسين |  |
|  | 20180381 | عمر محمد أبو رحاب |  |
|  | 20180054 | احمد عبد الله محمد |  |
|  | 20180320 | عبد الرحمن عصام عباس |  |
|  |  |  |  |
|  |  |  |  |

Delivered to:

**Dr. Wessam El-Behaidy**

**Eng. Salma Doma**

**Eng. Ahmed Nady**

* **Paper Details**
* Paper Name: Multi-Class Image Classification using CNN and TF-lite
* Authors: Vishal Shah1 and Neha Sajnani

International Journal of Research in Engineering, Science and Management Volume-3, Issue-11, November-2020https://www.ijresm.com | ISSN (Online): 2581-5792

* **Project Description**

Image classification is widely used in the field of machine learning nowadays

In the process of the depth convolutional network, it is obvious that the ability of the network will enhance as we adding up layers [3].As we are adding more and more layers it will gain the accuracy of the network and correctly classify into given classes.

Based on this idea, we built a simple convolutional network to classify the images into multi classes. For that, we took the Rock-Paper-Scissors dataset to test our convolutional network. In this dataset, we have 2520 training images and 372 testing images. We further divided the training images into validation images and then take 3 convolutional layers to train the model on those images.

# **Datasets for it** :[**Rock Paper Scissors Dataset**](https://www.kaggle.com/datasets/sanikamal/rock-paper-scissors-dataset)

It includes 2925 files for (Training & Testing)

## **Training (**3 directories)

* Rock
* paper
* Scissors

## **Testing (**3 directories)

* Rock
* paper
* Scissors

dimension of images = 300\*300px

* **Implementation details**

Then we split training data to (training and validation) with 20%

Now we have :

* Training (2016)
* Validation(504)
* Testing (372)
* **Our Model**

Diagram

Description automatically generated

* **Testing results**

Before the optimization

**Chart, line chart

Description automatically generated with medium confidence** **Chart, line chart

Description automatically generated**

After the optimization

**Chart, line chart

Description automatically generated** **Chart, line chart

Description automatically generated**

**Accuracy: 92,5 %**

**Graphical user interface, text, application

Description automatically generated**