Muğla Sıtkı Koçman University

Department of Computer Engineering

Senior Design Project I

RMDC(RainManDetection&Classification), Image Editor

Analysis & Design Report

Yasin ORTA

Supervisor: Zeynep Filiz Eren

January 18, 2020

**Contents**

1 Introduction 3

2 Motivation 3

3 Similar Existing Applications 3

4 Proposed System 3

4.1 Overview 3

4.2 Functional Requirements 3

4.3 User Interface Design 3

4.4 Entity Relationship Diagram 3

4.5 Technologies and Tools 3

5 Future Work 3

6 References 3

RMDC(RainManDetection&Classification)

# Introduction

Object detection and classification are important tools in computer vision that involves identifying and locating objects within the given image or video. As technology improves further and further it enhances many areas in business life. One example is logistic department, object detection is used to aid logistics in counting, identifying, and giving certain information. Even though there are applications which focus on this approach there is still room for improvement.

This project aims to improve many fields such as logistic efficiency by developing RMDC, a user-friendly application that will find desired objects in the given inputs from users by analyzing them and producing accurate and fast results. RMDC is not limited to logistics and it can provide results to anybody on any objects accurately and fast.

# Motivation

The motivation for this project comes from the fact that similar applications on the market are hard to reach and hard to use as well as address certain individuals while RMDC focuses on all groups of users, let it be for logistics or simple school activities to medical good counting.

RMDC strives to help every user that requires counting, classification, and analysis by offering following benefits;

* Not limited to any field or group of users, anyone with counting and classification problem can use it.
* Producing fast and accurate results.
* Detailed or simple interaction depending on the user.

# Similar Existing Applications

Object Counter By Camera from Ocean Float Mobile; Here is the Object Counter By Camera App. You can count things with this counting app via the camera in a shot.

JetCounter By Variance InfoTech Pvt Ltd; JetCounter Mobil Uygulaması ile şu anda reklamsız ücretsiz olarak 'n' sayıda nesne sayın.

Object Counter-count things By Xiaozhao Inc; İşte Kamera Uygulamasıyla Nesne Sayacı. Bu sayma uygulamasıyla bir çekimde kamera aracılığıyla bir şeyleri sayabilirsiniz. örnek şeyler: tablet, haplar, borular, madeni paralar, kutular, çubuklar, kereste

In addition to using camera, RMDC can use image inputs from other sources to analyze and produce results.

RMDC requires simple interactions and can be used by any users for any objects.

# Proposed System

## Overview

Data: Image input which can be acquired from camera, video or simply by loading from gallery.

Process: Machine model that will analyze and produce results on the given input.

Display: Showing results to the user.

A diagram with a black text

Description automatically generated with medium confidence

## Functional Requirements

A diagram of a diagram

Description automatically generated

Description;

* **Classify Image**: Inputs will be checked whether they have the objects user specified or not.
* **Count Objects**: Inputs will be processed and desired objects will be counted.
* **Save Result along with image**: New input image will be created and results will be written to it and saved to the output file.

## User Interface Design

## Entity Relationship Diagram

## Technologies and Tools

* TensorFlow Machine learning algorithm
* Yolo Object Detection System
* Jupyter Notebook
* Python Language
* Depending on the progress of the project ReactNative for application part

# Future Work

* Currently, tensorflow algorithm is used and with annotations accuracy is close to %90, this will be improved.
* Yolo object detection system will be implemented to compare the performance.
* User Interface and Entity Relationship diagrams will be designed.
* By using ReactNative an application will be created.
* Performance improvements will be made to CNN architecture.

# References

* <https://play.google.com/store/apps/details?id=com.objectcounter.utility.app2022&hl=en_US>
* <https://play.google.com/store/apps/details?id=com.jetcounter>
* <https://play.google.com/store/apps/details?id=xz.camera.objectcounter>
* <https://www.tensorflow.org/>
* <https://github.com/ultralytics/ultralytics>
* [https://lucid.app/lucidchart/ff0ba659-cd67-44c0-a031-5f81afa2b2fa/edit?beaconFlowId=5C66747C210B364B&invitationId=inv\_80506765-85ae-42f8-8688-04838cc72df5&page=0\_0#](https://lucid.app/lucidchart/ff0ba659-cd67-44c0-a031-5f81afa2b2fa/edit?beaconFlowId=5C66747C210B364B&invitationId=inv_80506765-85ae-42f8-8688-04838cc72df5&page=0_0)

Image Editor

# Introduction

Image processing is a multidisciplinary area that has became something of indispensable aspect of modern technology that can transform, analyze, and interpret visual information. It is important to ready the data for the analysis which requires some certain steps.

This project aims to reduce these steps and make data preparation steps easier by developing Image Editor that will edit given images with simple commands effectively and fast.

# Motivation

The motivation came from a need from another project called Allerview, to prepare required input data in specific ways. This application uses python language and several libraries and some scripts to handle several image processing requirements.

# Similar Existing Applications

Adobe

Photo Editor: BeFunky

Photopea

While there are many applications that can take care of image editing, this project born from the need of another project and while most of the applications can handle some of these procedures, our project is a more specialized version of them.

# Proposed System

## Overview

A diagram with a black text

Description automatically generated with medium confidence

Input files are given through specifying the location of the file.

There are several methods which uses different algorithms and approaches such as adaptive threshold, parameter manipulation with thresholding, HSV color editor, morphology and region of interest.

## Functional Requirements

A diagram of a software development process

Description automatically generated

Adaptive Threshold: This method finds certain areas using averaged parameters thus eliminating the parameter manipulation.

HSV Color Editor: By setting correct boundaries for hsv color channels, algorithm finds the results.

Morphology: By applying said methods, outputs details are highly enhanced.

Region of interest: By cutting only the interested areas, process time can be decreased enormously.

## User Interface Design

List UI Mockups and their descriptions

Provide a youtube link to Wizard of Oz Video

## Entity Relationship Diagram

Provide ER Diagram and describe eaach table and columns in the diagram

## Technologies and Tools

* OpenCV Library
* Python Language
* Numpy Library

# Future Work

Currently this project is only usable as a script but;

* By using ReactNative an application can be created specifically for editing input images.
* Currently shadow removal is an obstacle that no method can handle it effectively, a research about it will be done.
* Depending on the requirements additional functional scripts will be written.
* Proposed system section will be filled out accordingly with the development of the project.

# References

* <https://opencv.org/>
* <https://numpy.org/>