

**MOBILE APPLICATION FOR IMAGE LABELING BASED ON  
DEEP LEARNING API**

**MUSTAFA FAWZI MUSTAFA**

**AYA HUSSEIN ALAEBY**

**UNIVERSITY OF INFORMATION TECHNOLOGY AND  
COMMUNICATIONS/ BUSINESS INFORMATICS  
COLLEGE/BUSINESS INFORMATION TECHNOLOGY**

**BAGHDAD  
2019**



# **MOBILE APPLICATION FOR IMAGE LABELING BASED ON DEEP LEARNING API**

**MUSTAFA FAWZI & AYA HUSSEIN**

**RESEARCH PROJECT REPORT SUBMITTED IN PARTIAL  
FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF  
THE DEGREE OF [BACHELOR]**

**UNIVERSITY OF INFORMATION TECHNOLOGY AND  
COMMUNICATIONS/ BUSINESS INFORMATICS  
COLLEGE/BUSINESS INFORMATION TECHNOLOGY**

**BAGHDAD  
2019**

**UNIVERSITY OF INFORMATION TECHNOLOGY AND  
COMMUNICATIONS  
ORIGINAL LITERARY WORK DECLARATION**

Name of Candidate: Mustafa Fawzi & Aya Hussien

Matric No:

Name of Degree: bachelor

Title of Project Research Report ("Mobile application for image labeling based on deep learning API")

Field of Study: mobile application

I do solemnly and sincerely declare that:

- (1) I am the sole author/writer of this Work;
- (2) This Work is original;
- (3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
- (4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
- (5) I hereby assign all and every rights in the copyright to this Work to the University of Information Technology and Communications ("UOITC"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UOITC having been first had and obtained;
- (6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UOITC.

Candidate's Signature

Date:

Subscribed and solemnly declared before,

Witness's Signature

Date:

Name:

Designation:

# **MOBILE APPLICATION FOR IMAGE LABELING BASED ON DEEPLARNING API**

## **ABSTRACT**

Connecting the physical and the digital world is an upcoming trend that enables numberless use-cases. The mobile phone as the most pervasive digital device is often used to establish such a connection. The phone enables users to retrieve, use, and share digital information and services connected to physical objects. Recognizing physical objects is thus a fundamental precondition for such applications. Object recognition is usually enabled using visual marker (e.g. Qr Codes) or electronic marker (e.g. RFID). Marker based approaches are not feasible for a large range of objects such as sights, photos, and persons. In this project, we present object recognition, optical character recognition, text translation and solve math equation mobile app that develops by Flutter framework and processes camera images per second and running on various platform (Android, iOS and windows) based on Deep learning model which is programmed by google cloud.

Keywords: Mobile Application, Deep learning, Flutter, Object Recognition.

## الخلاصة

ربط العالم المادي بلعالم الرقمي هو الاتجاه القادم لحل كثير من الحالات ، ولان الهاتف المحمول هو اكثر الأجهزة الرقمية شيوعاً واستخداماً عادة ما يستعمل وهكذا غرض ، الهاتف المحمول يتيح للمستخدم استرجاع واستخدام و مشاركة المعلومات الرقمية والخدمات المتصلة بلاشياء المادية . التعرف على الأشياء المادية هو حجر الأساس لمثل هذه التطبيقات ، التعرف على الأشياء ممكن عادة بأستخدام العلامة الالكترونية او بطاقة التعريف الالكترونية ، أساس هذه الطرق ليس قادر على التعرف على عدد واسع من الأشياء مثل الصور او الناس .

في هذا المشروع نحن نقدم تطبيق موبايل للتعرف على الكائنات وتتميز الاحرف وترجمة النصوص وحل المعادلات الرياضية تم تطويره عن طريق بيئة فلاتر ويعالج صور الكاميرا خلال ثانية واحدة وله القدرة على العمل على مختلف المنصات مثل اندرويد ونظام iOS ونظام الويندوز بلاعتماد على نموذج للتعلم العميق تم برمجته عن طريق كوكل كلاود .

## **ACKNOWLEDGEMENTS**

First and foremost, I would like to thank God for his blessings and giving me the power to believe in my passion and pursue my dreams. Because of him, I were able to finish this dissertation without being ill, or facing any obstacles. I thank him for the gift of life, health and all life opportunities. I would like to express my deepest gratitude to my supervisor Asst. Dr. Mohammed Salih Mahdi and Asst. Ahmed Raheem for their guidance, encouragement and advice throughout the work on this project.

I would like also to thank all friends and especially the friends who supported me in any way during the creation period of this project.

## TABLE OF CONTENTS

Abstract .....	i
الخلاصة.....	ii
Acknowledgements.....	iii
Table of Contents.....	iv
List of Figures.....	vi
List of Tables .....	vii
List of Symbols and Abbreviations .....	viii
<b>CHAPTER 1: GENERAL INTRODUCTION.....</b>	<b>1</b>
1.1 Introduction of mobile computing .....	1
1.2 Mobile applications .....	1
1.2.1 Native vs Hybird .....	3
1.3 Artifical Neural Network .....	4
1.3.1 Areas and characterization of ANN .....	5
1.3.2 Properties of ANN .....	5
1.3.3 Architecture of ANN .....	6
1.3.4 Convolutional Neural Network(CNN) .....	7
1.4 The problem of project .....	8
1.5 The aim of project .....	8
1.6 Project outline .....	8
<b>CHAPTER 2: PRIVOUS WORK.....</b>	<b>9</b>
2.1 Related work .....	9
<b>CHAPTER 3: METHODOLOGY .....</b>	<b>12</b>
3.1 Object Recognition Mobile App (ORMA) .....	12
3.1.1 Read Images from Google Dataset .....	12
3.1.2 Deep Learning API .....	13
3.1.3 Recognition Model .....	13
3.2 Mobile Application Flutter Framework .....	14
3.2.1 Dart Platform .....	14
3.2.2 Widgets .....	15
3.2.3 Flutter Features .....	15

3.2.4	Flutter benefit .....	15
3.3	Firestore .....	17
3.3.1	Firestore MLkit .....	18
<b>CHAPTER 4: RESULT .....</b>		<b>20</b>
4.1	Implimntation of ORMA .....	20
<b>CHAPTER 5: CONCLUSIONS AND RECOMMENDATIONS .....</b>		<b>25</b>
5.1	Conclusions .....	25
5.2	Recommendations .....	25
<b>REFREANCES .....</b>		<b>26</b>



## LIST OF FIGURES

Figure 1.1: Native App development structure .....	2
Figure 1.2: Hybrid App development structure .....	3
Figure 1.3: A single-layer Neural Net .....	6
Figure 1.4: A Multilayer Neural Net .....	6
Figure 3.1: Phase structure of the phase ORMA .....	12
Figure 3.2: Deep learning Kit using convolution neural network (CNN).....	13
Figure 3.3: flutter advantages .....	17
Figure 3.4: Firebase .....	17
Figure 3.5: MLkit features .....	18
Figure 3.6: MLkit APIs .....	19
Figure 4.1: First Interface of ORMA .....	20
Figure 4.2: Interface Select Image .....	21
Figure 4.3: Text recognition .....	21
Figure 4.4: Object recognition .....	22
Figure 4.5: Solve math equation .....	22
Figure 4.6: Text translation .....	23
Figure 4.7: Shared button .....	24
Figure 4.8: Share interface.....	24

## LIST OF TABLES

Table 1.1: Native vs Hybrid .....	3
Table 1.2: React vs Ionic vs Flutter .....	4

## List of Symbols and Abbreviations

CCN	:	Convolutional Neural Network
ANN	:	Artificial Neural Network
RFID	:	Radio-frequency identification
AOT	:	Ahead-of-time
FCM	:	Firebase Cloud Messaging
GCM	:	Google Cloud Messaging

## CHAPTER 1: GENERAL INTRODUCTION

### 1.1 Introduction of Mobile Computing

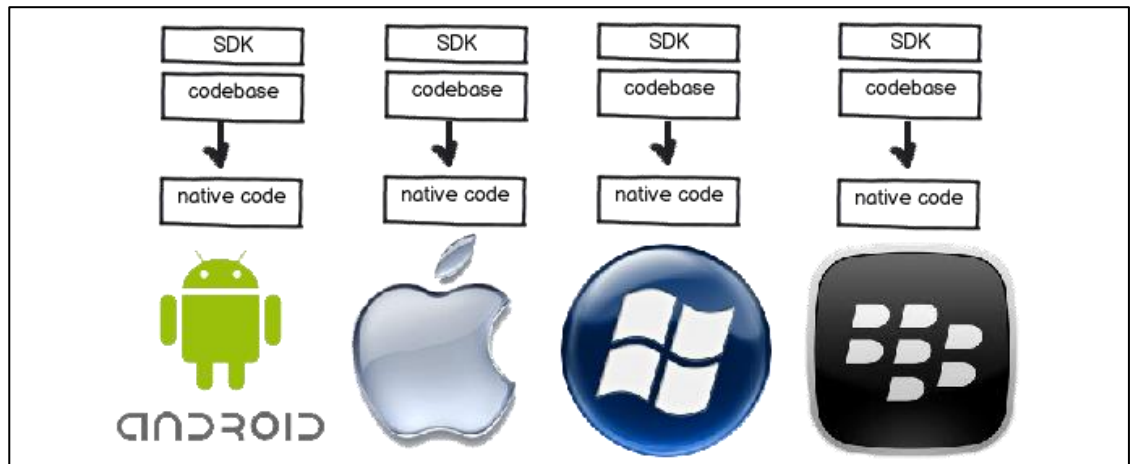
The term "Mobile computing" is used to describe the use of computing devices, which usually interact in some fashion with a central information system--while away from the normal, fixed workplace. A technology allows transmission of data, via a computer, without having to be connected to a fixed physical link. Mobile computing technology enables the mobile worker to create, access, process, store and communicate information without being constrained to a single location. By extending the reach of an organization's fixed information system, mobile computing enables interaction with organizational personnel that were previously disconnected. It provides the continuous access to the wireless network services and the flexible communication between the people. It provides the real-time business to employee communication, enhanced customers interactions, and fastest communication between the individuals. The communication occurs with the real-time wireless connection. It provides the data, audio and video access to any user, any time with a wireless enable device[1]. Furthermore, the name MOBILE is derived from the first letter in each of the six categories that make up the framework. The six categories are[2]:

- M the need for mobility.
- O the need to improve operations.
- B the need to break business barriers.
- I the need to improve information quality.
- L the need to decrease transaction lag.
- E the need to improve efficiency.

### 1.2 Mobile Application

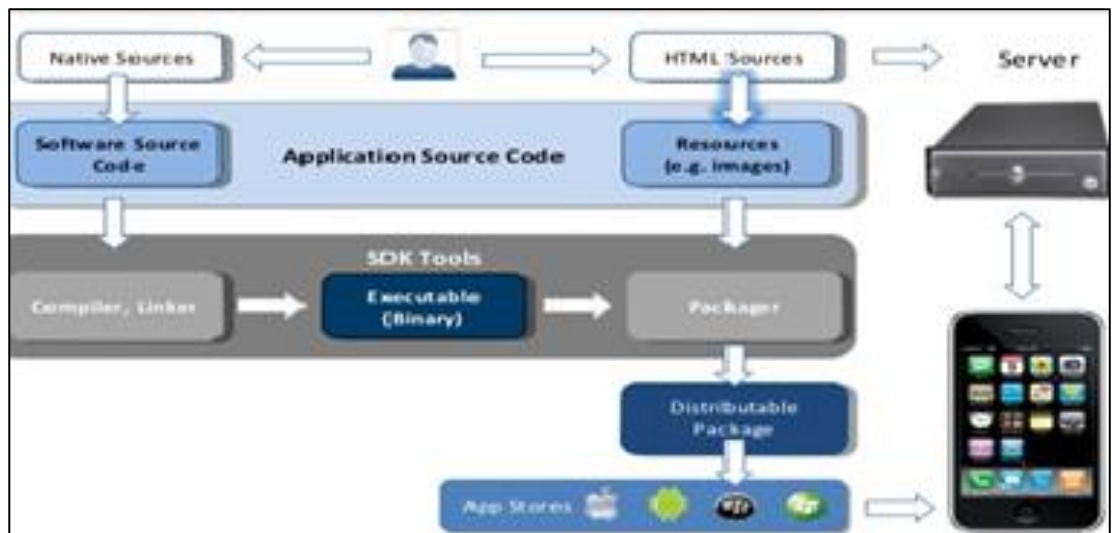
In Today's Generation are much more depended on mobile applications. Mobile application so, the Mobile application development are very essential to any business organization. We want the fast application development in today's environment. There are different kinds of ways to develop the mobile apps. Also, there are different frameworks to develop the mobile apps. Businesses that need a mobile app have three available options [3]:

- Native apps can Built using the device's native programming language, native apps only run on their designated platforms as shown in figure 1.1. For example, Android applications can't run on iOS and Blackberry applications can't run on Android, iOS applications can't run on Windows platform, and so on. Native apps are distributed through their respective platforms with marketplace/app store and installed on the devices itself.



**Figure 1. 1 Native App development structure**

- Hybrid apps: Hybrid apps are the cross platform native apps and mobile web apps. A hybrid app is a mobile app wrapped in a platform-specific shells. This platform-specific shell gives the application native qualities, such as full device integration, Portability, native installation, and app store/market distribution [3]. Hybrid app development structure show in figure 1.2.



**Figure 1. 1 Hybrid App development structure**

### 1.2.1 Native Vs Hybrid Apps

In this section we provide an explanation of how native and hybrid apps presumably address key characteristics of apps usage and development shown in table 1.1 and comparison between React native , Ionic and Flutter as shown in table 1.2 [3] .

**Table 1. 1 native vs hybrid [3]**

Hybrid app	Native app
Developed for cross platform , ionic or react native ,etc.	Developed in platform specific language, objective-c or swift for iOS, java for android, etc.
Write once, run anywhere	Separate code for each platform
Save time and money	Higher investment of time, talent and resources
Faster development cycle	Higher costs and development time

**Table 1. 2 React vs Ionic vs Flutter [3]**

Attribute	React native	Ionic	Flutter
Programming language	JavaScript , Swift , Java or Objective-C	Html , Css , Typescript	Dart
Performance	Close to native	Moderate	Amazing
GUI	Use native UI controllers	Html , Css	Use proprietary widgets and deliver amazing UI

### 1.3 Artificial Neural Network

ARTIFICIAL NEURAL NETWORK (ANN) is one of artificial intelligence models that have been studied for many years with the hope of achieving "Human-like performance", Different names were given to these models such as [4]:

- Parallel distributed processing models
- Biological computers or Electronic Brains.

After that, all these names settled on Artificial Neural Networks (ANN) and after it on neural networks (NN) only.

There are two basic different between computers and neural, these are:

- 1- These models are composed of many non-linear computational elements operating in parallel and arranged in patterns reminiscent of biological neural networks.
- 2- Computational Elements (or node s) are connected via weights that are typically adapted during use to improve performance just like human brain.

Computer —————> logic Elements (1, 0)

Neural —————> weighted performance

An artificial neural network is an information processing system that has certain performance characters in common with biological neural networks. Artificial neural networks have been developed as generalizations of mathematical models of human cognition or neural biology, based on the assumptions that:-

1. Information processing occurs at many simple elements called neurons.
2. Signals are passed between neurons over connection links.
3. Each connection link has an associated weight which, in a typical neural net, multiplies the signal transmitted.
4. Each neuron applies an action function (usually nonlinear) to its net input (sum of weighted input signals) to determine its output signal.

### **1.3.1 Areas and characterization of ANN**

The areas in which neural networks are currently being applied are [5]:

1. signal processing
2. Pattern Recognition.
3. control problems
4. medicine
5. speech Recognition
6. Business

A Neural network is characterized by [6]:

1. Architecture: - its pattern of connections between the neurons.
2. Training Learning Algorithm: - its method of determining the weights on the connections.
3. Activation function.

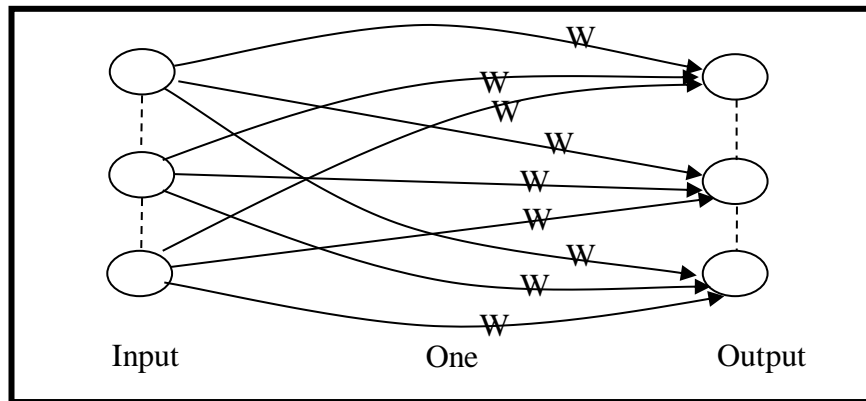
### **1.3.2 Properties of ANN**

- Parallelism
- Capacity for adaptation "learning rather programming"
- Abstraction & solving problem with noisy data.
- Ease of constriction & learning.
- Fault tolerance



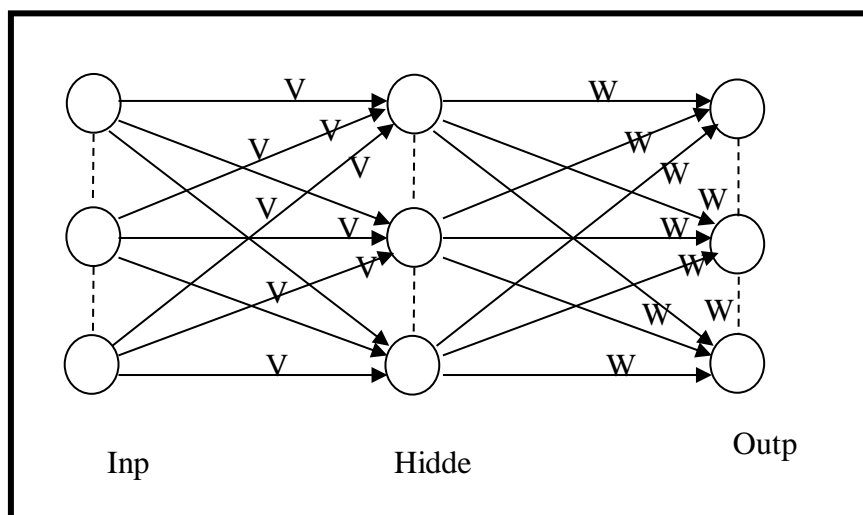
### 1.3.3 Architecture of ANN

**Single-Layer Net:-**A single-layer net has one layer of connection weight. Often, the units can be distinguished as input units, which receive signals from the outside world, and output units, from which the response of the net can be read as shown in figure 1.3 [7]



**Figure 1.3: A single-layer Neural Net**

**Multilayer net** A Multilayer net is a net with one or more layers (or levels) of nodes which is called hidden units, between the input units and the output units. Typically, there is a layer of weights between two adjacent levels of units (input, hidden, or output). Multilayer nets can solve more complicated problems than can single-layer nets, but training may be more difficult. However, in some cases, training may be more successful because it is possible to solve a problem that a single-layer net cannot be trained to perform correctly at all. The Figure 1.4 shows the multilayer neural net [7].



**Figure 1.4: A Multilayer Neural Net**

### 1.3.4 Convolutional Neural Network (CNN)

CNN is a class of deep, feed-forward artificial neural networks, the most important characteristics of CNN is the local connectivity and the using of shared weights; thus, it can learn local features of the input image [8].

In image classification, the CNN is very effective and fast method in recognition phase, even in partly rotation, distortion, scaling and translation variations, but it is slow in training phase, so training some thousands of images could take several hours.

Back propagation is used for training the CNN by adjusting the weights of filters (kernels). The weights are updated using Eq. (1.1):

$$(w_{jk})_{\text{new}} = (w_{jk})_{\text{old}} - \eta * \Delta w_{jk} \dots\dots\dots (1.1)$$

Where  $w_{jk}$  is the weights of the filters,  $\eta$  is the learning rate equal to 0.0001, and  $\Delta w_{jk}$  is the derivative of error regarded to weights.

The typical CNN structure consists of the following layers [9]:

- **Convolutional Layers:** takes several feature maps as input and produce  $n$  of feature maps as output, where  $n$  is the number of learnable filters in the convolution layer. Filter weights are adapted using back propagation depending on training data. The number  $n$  of filters as well as the filter's size  $k_w \times k_h$  are hyper parameters of convolutional layers. The outputs of this layer go through Rectified Linear Unit (ReLU) that is the common choices for non-linear activation function.
- **Pooling Layers:** Pooling summarizes a  $p \times p$  area of the input feature map. Pooling can be used with a stride of  $s \in \mathbb{N} \geq 2$ , pooling layers are sometimes also called *subsampling layers* is applied for three reasons: To get local translational invariance, to get invariance against minor local changes and, most important, for data reduction to  $(1/s^2)$ th of the data.
- **Fully-Connected Layer:** Fully Connected layer is a traditional multi-layer perceptron where every neuron in a layer is connected to each neuron on the following layer. The work of the fully connected layer is to classify the features extracted by previews (convolutional and pooling layers) to the proper classes. Dropout technique can be applied over this layer, it is used to prevent over fitting by setting the output of any neuron to zero with probability  $p$ .

#### 1.4 The problem of project

Approaches to sense real world objects are usually based on visual markers (e.g. QR-Codes or other 2D barcodes RFID tags). For certain types of objects, such as sights, buildings, and living beings marker based approaches are often not sensible or considerably restrict the interaction radius.. Thus, performed on the mobile device with a delay of up to several seconds. Either way this delay clearly restricts the interaction. However, the current Artificial Intelligence models, which is, recognize specific object need to a huge quantity of memory and making waste of power that indicates as consumption in power of smart phone.

#### 1.5 The Aim of the Project

The aim of this project is to design and implement mobile app develops by Flutter framework and processes camera images per second and running on various platform (Android, IOS and windows). Furthermore, it has ability to recognize a huge amount of objects up to 8000 objects, text recognition, text translation and solve math equations based on deep learning neural network model which is programed by google cloud.

#### 1.6 Project Outline

The Project consists of five chapters. The first chapter is an introduction. The remaining chapters are as follows:

- *Chapter Two* presents the background of previous works
- *Chapter Three* introduces the project tools, the structure of the proposed.
- *Chapter Four* introduces the description of object recognition mobile application, including their environment, components, and accuracy.
- *Chapter Five* gives the conclusions of the project and presents recommendations for future work.

## CHAPTER 2: PREVIOUS WORK

### 2.1 Related Works

There are numerous of studies have been analyzed from the scientific literature about object recognition and optical character recognition based on artificial intelligence techniques and set of operations as follows:

- Ahmed Q. AL-Thahab,2015 [10] In this paper, classification method based on multiwavelet transform and radon transform that proposed, and these two transforms combine together to extract useful information from image, and then forward these features extraction for classification by using robust method of artificial neural network. The aim of this paper is that how the noisy image can be classified properly into original image via high recognition rate.
- Laine M. and Olli S., 2016, [11] made a system that just deals with the English capital letters. In the beginning, the input captured mage is skew. This image is repaired by searching for a line with the greatest number of successive white pixels through maximizing the given alignment standard. After that, segmentation is done for the image on the basis of X-Y Tree decomposition then the image is to be recognized via detecting the Manhattan distance based similarity for a set of centroid to boundary features.
- Anwar Hassan Mahdy,2016[12] This research proposed pattern recognition system consists of two-stage process. The first stage is feature extraction and the second stage is classification. Feature extraction is the measurement on a population of entities that will be used in recognition process. This assists the recognition stage by looking for features that allows fairly easy to distinguish between the different classes. Several different features have been used for recognition process. The set of proposed features that are used makes up a feature vector. These set of features are: the first feature is represented the number of character pixels (the summation of pixels), the second features is represented the width of each character in pixels, and the third feature represented the height of each character in pixels. Finally, Pattern recognition system classifies each member of the population on the basis of information contained in the feature vector. The results show that the suggested features gives higher accuracy in text and character recognition.

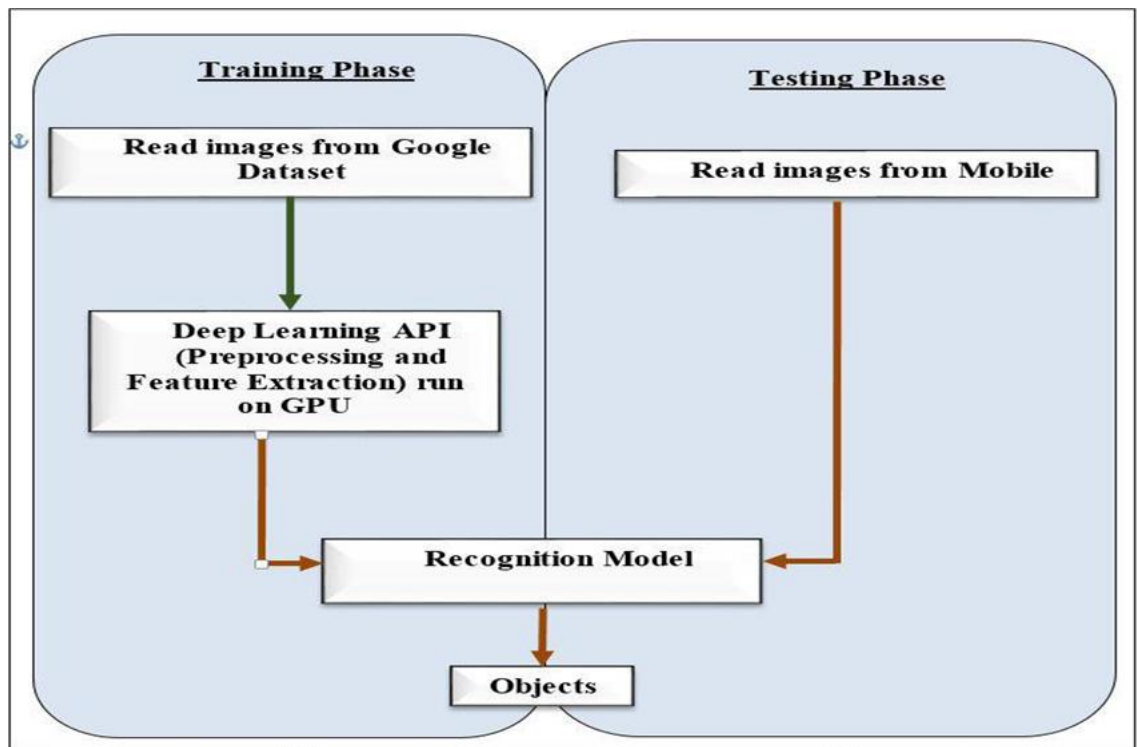
- Lucas P ,2017 [13] , the presented work provides a generic technology for the recognition of urban objects, i.e., buildings, in terms of a reliable mobile vision service using android studio. The presented recognition system from local descriptor responses in the mobile imagery. This paper present an improvement over standard image descriptors by selecting only informative keys for descriptor matching.
- Román Osorio, N. Bustos,2017,[14]. In this article, the implementation of a machine vision algorithm into a mobile navigation robot for object recognition is presented. By using its color and machine vision algorithms, such as Canny and the Hough transform as a complement in order, improves the recognition of the selected object. These algorithms is implemented in a device with Android Operating System as an application.
- Wen-Yen Tseng, Kai-Hsiang Chen,2018 [15]. In this study, developed a system based on a mobile phone App for the collection of information pertaining to the location of objects in images. The proposed system is simple and easy to use. Experiments demonstrate the excellent performance of the proposed system with regard to accuracy and response time. This study demonstrates the feasibility of collecting image information using mobile phones.
- Yue Liu, Ju Yang, Mingjun Liu, 2018[16], Quick Response Code has been widely used in the automatic identification fields. In order to adapting various sizes, a little dirty or damaged, and various lighting conditions of bar code image, this paper proposes a novel implementation of real-time Quick Response Code recognition using mobile, which is an efficient technology used for data transferring. An image processing system based on mobile is described to be able to binaries, locate, segment, and decode the QR Code.
- Chucai Yi, and Yingli Tian, 2018[17], this paper proposes a method of scene text recognition from detected text regions. In text detection, proposed algorithms are applied to obtain text regions from scene image. First, designing a discriminative character descriptor by combining several state-of-the-art feature detectors and descriptors. Second, model character structure at each character class by designing stroke configuration maps. An Android is developed to show the effectiveness of proposed method on scene text information extraction from nearby objects. The demo system also provides us some insight into algorithm design and performance improvement of scene text extraction.

- Weinman et al 2018. [18] combined the Gabor-based appearance model, a language model related to simultaneity frequency and letter case, similarity model, and lexicon model to perform scene character recognition

## CHAPTER 3: METHODOLOGY

### 3.1 Object Recognition Mobile App (ORMA)

The stages of the structure of the ORMA is consists of two stages: the training and the testing stage. Each stage has specific functions, all the functions are explained in detail in the following subsections. Figure (3.1) describe the block diagrams of the training and testing stages of ORMA respectively.



**Figure (3.1): Phase structure of the phase ORMA: the main block diagram of training set of ORMA and the main block diagram of testing of ORMA**

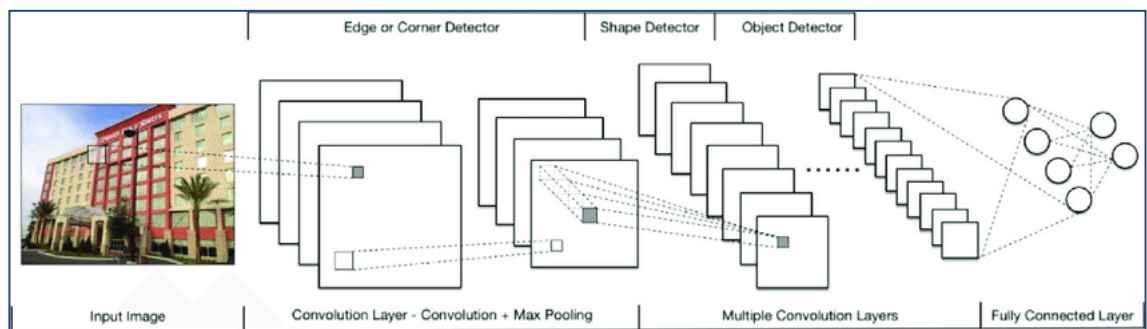
#### 3.1.1 Read Images from Google Dataset

The Google Dataset used in this project were taken from Google cloud , related to google Kit's image labeling APIs, you can recognize entities in an image without having to provide any additional contextual metadata, using either an on-device API or a cloud-based API.. Millions of images in this project were selected from the Google Dataset. The format of these images are JPG, PNG, BNP and GIF with different size. The device-based API supports 8000 labels, such as the following examples: Activities (Dancing Eating, etc.), Things (car, piano, etc.), Animals (dog, cat, tiger, etc.), places (beach, mountain), etc.

### 3.1.2 Deep learning API

Deep learning Kit is a mobile SDK that brings Google's machine learning expertise to Android and iOS apps in a powerful yet easy-to-use package. Whether you're new or experienced in Deep learning, you can implement the functionality you need in just lines of code. There's no need to have deep knowledge of neural networks or model optimization to get started. On the other hand, if you are an experienced Deep learning developer, Deep learning Kit provides convenient APIs that help you use your custom TensorFlow Lite models in your mobile.

Preprocessing and feature extraction has been achieved by Deep learning Kit using convolution neural network(CNN) as shown in figure 4.2 ,where input image has been entered to CNN. Edge detector, shape detector and object detector has been implemented based on convolution layer and max pooling. The last layer is fully connected layer.



**Figure 3.2: Deep learning Kit using convolution neural network (CNN)**

### 3.1.3 Recognition Model

The recognition model is built based on Google Dataset which has trained by Deep learning Kit. Mobile images are test by recognition model and processes per second.



## 3.2 Mobile Application Flutter Framework

Flutter Framework is an open-source mobile application development framework created by Google. It is used to develop applications for Android, iOS, Web, Windows, Linux, as well as being the primary method of creating applications for Google Fuchsia. Flutter framework which used for the designing and implement the application.

### 3.2.1 Dart platform

Flutter apps are written in the Dart language and make use of many of the language's more advanced features. On Android, and on Windows, macOS and Linux via the official Flutter Desktop Embedding project. A notable feature of the Dart platform is its support for "hot reload" where modifications to source files can be injected into a running application.

Dart is a client-optimized programming language for fast apps on multiple platforms. It is developed by Google and is used to build mobile, desktop, backend and web applications. Dart is an object-oriented, class defined, garbage-collected language

#### **There are three main ways to run Dart code:**

1. Compiled as JavaScript: To run in mainstream web browsers, Dart relies on a source-to-source compiler to JavaScript. When running Dart code in a web browser the code is precompiled into JavaScript using the dart2js compiler.
2. Stand-alone: The Dart software development kit (SDK) ships with a stand-alone Dart VM, allowing Dart code to run in a command-line interface environment.

#### **Advantages and features of Dart**

1. Dart has a strong support base with many libraries and tools, which enable very large applications.
2. One of the major objectives of Dart is to simplify programming tasks. It is designed to make common programming tasks simpler.
3. Dart is very stable and it can be used to build production quality real-time applications. It is an object-oriented programming language with support for inheritance, interfaces and optional typing features.

### 3.2.2 Widgets

UI design in Flutter typically involves assembling and/or creating various widgets. A widget in Flutter represents an immutable description of part of the user interface; all graphics, including text, shapes, and animations are created using widgets. More complex widgets can be created by combining many simpler ones.

However, the use of widgets is not strictly required to build Flutter apps. An alternative option is to use the Foundation library's methods directly, interfacing with "canvas" commands to draw shapes, text, and imagery directly to the screen.

### 3.2.3 Flutter features

- It is open-source
- It is dart based (Dart is a famous object-oriented programming language)
- It offers widgets that can be customized as per specific designs (Flutter has two kinds of widgets- Material Design widgets and Cupertino widgets)
- It bases its architecture on reactive programming
- It is a stiff rival to React Native

### 3.2.4 Flutter benefit

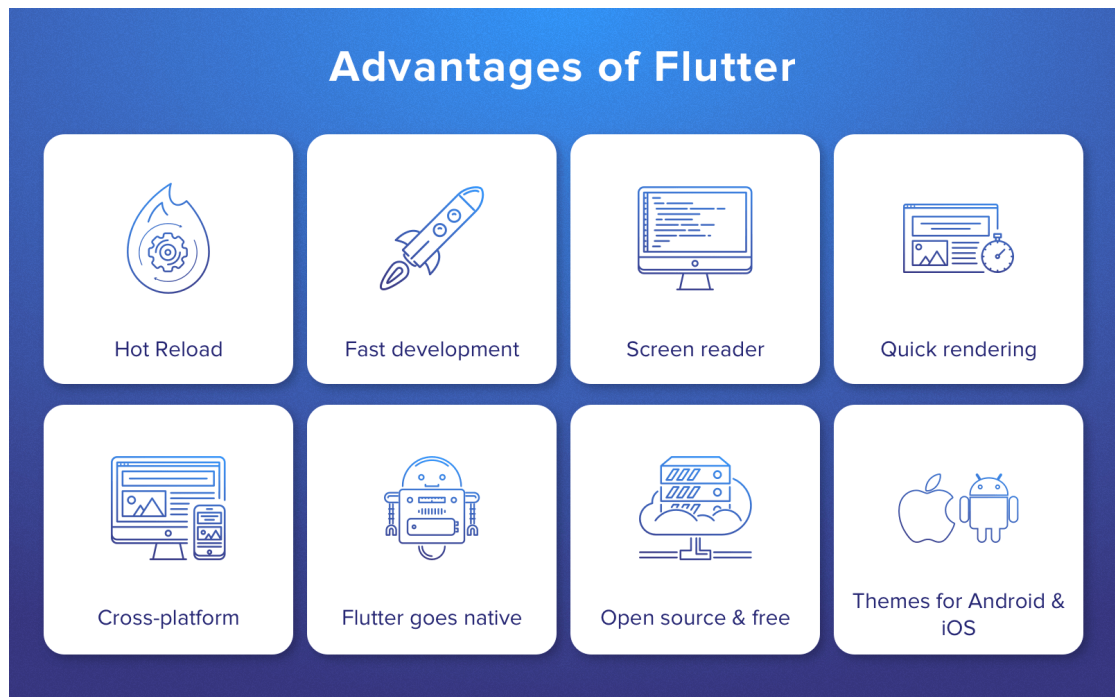
From the perspective of an App Owner, the crucial advantages are thus: Flutter speeds up the mobile app development process, reduces the cost of app production, and helps your team to build a beautiful app UI with smooth animations. Let's take a look at it more deeply.

- 1. Faster code writing:** For developers, Flutter means faster & more dynamic mobile app development. We can make changes in the code and see them straight away in the app! This is the so-called hot reload, which usually only takes (milli) seconds and helps teams add features, fix bugs and experiment faster.

Hot reload is also very comfortable in developer-designer cooperation when we want to improve or experiment with an app's look and check the effects on the spot. In other words, with Flutter, your designer or tester can work together with

a developer on the UI, making changes – for example, “Put it 2 pixels right” or “Make the animation faster” – and see them immediately.

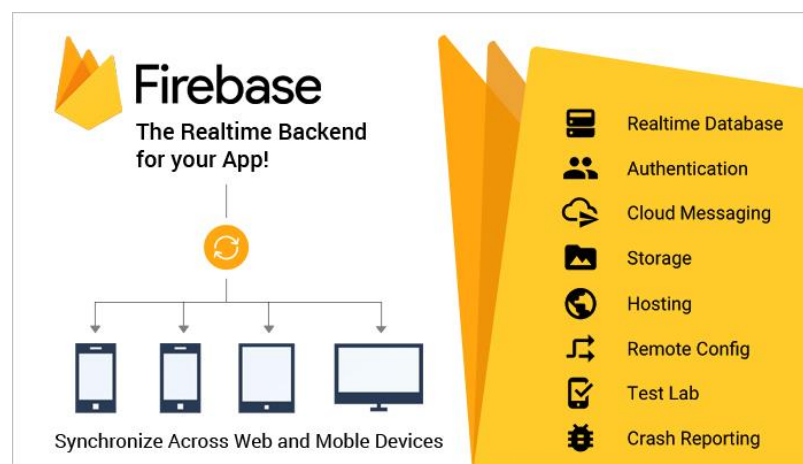
2. **One code for many platforms:** Developers write just one codebase for your many apps – covering to Android, iOS, Web, windows and Linux platforms. Flutter doesn’t depend on the platform, because it has its own widgets and designs. This means that you have the same app on many platforms. Yet what’s important is that, if you want to differentiate your apps – it’s possible.
3. **Less testing:** If you have the same app for many platforms, it means less testing! The Quality Assurance process can be faster. Because of one codebase, the developers write automatic tests only once. What’s more, Quality Assurance specialists have less work to do, because they have only one app to check. Of course, if your apps have some differences, they need to be tested on both platforms.
4. **Faster apps:** Flutter apps work in a smooth and fast way, without hanging and cutting while scrolling.
5. **Designs which your users will love:** Flutter is designed to make it easy to create your own widgets or customize the existing widgets. Here. You can browse a catalog of Flutter’s widgets and view, for example, Material Design widgets and Cupertino widgets.
6. **The same app UI on older devices:** Your new app will look the same, even on old versions of Android and iOS systems. There are no additional costs for supporting older devices. Flutter runs on Android Jelly Bean or newer, as well as iOS 8 or newer.
7. **Perfect for MVP:** Do you need an MVP (Minimum Viable Product) for your app to show it to your investors? Flutter will be perfect, especially if you have little time.



**Figure 3.3 flutter advantages**

### 3.3 Firebase

Firebase is a Backend as A Service or BAAS, meaning it is a NoSQL data store that is in the cloud that your clients can access directly in Real time. Now this means Firebase has its own Real time Database, Cloud Storage, Authentication, Hosting, MLkit and many other services. The Firebase Real time Database is a cloud-hosted database. Data is stored as JSON and synchronized in Real time to every connected client.

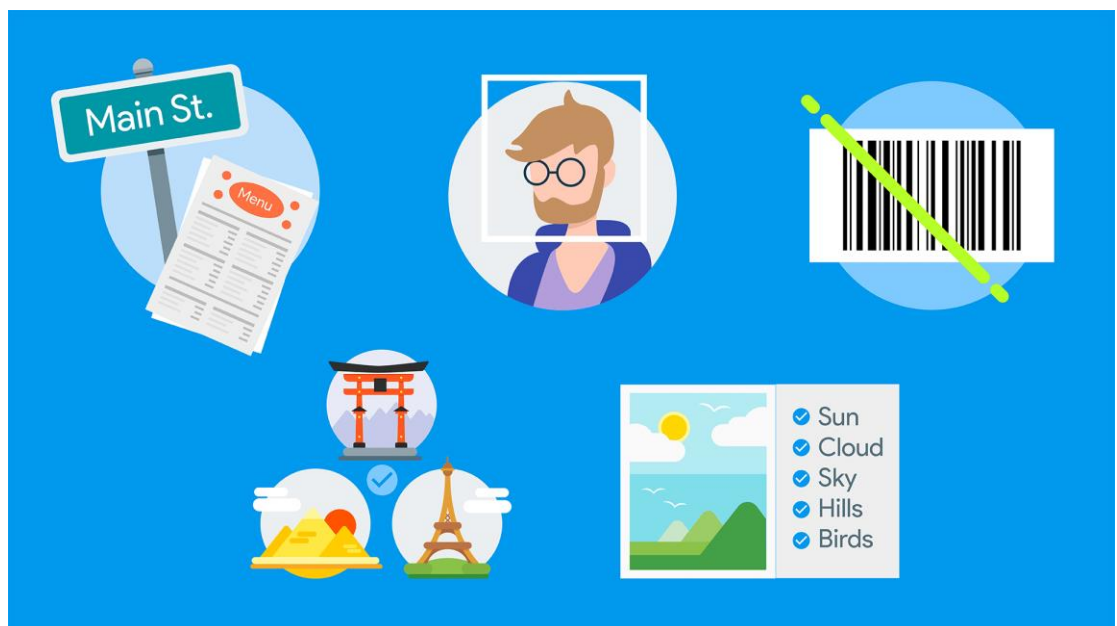


**Figure 3.4 firebase**

### 3.3.1 Firebase MLkit

Use Machine -Deep learning in your apps to solve real-world problems. ML Kit is a mobile SDK that brings Google's machine learning expertise to Android and iOS apps in a powerful yet easy-to-use package. Whether you're new or experienced in machine learning, you can implement the functionality you need in just a few lines of code. There's no need to have deep knowledge of neural networks or model optimization to get started. On the other hand, if you are an experienced ML developer, ML Kit provides convenient APIs that help you use your custom TensorFlow Lite models in your mobile apps.

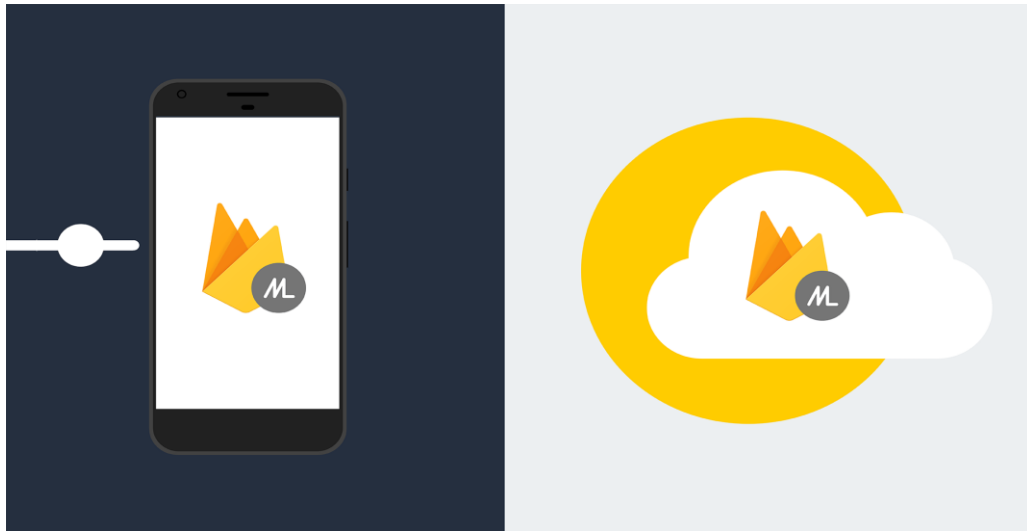
ML Kit comes with a set of ready-to-use APIs for common mobile use cases: recognizing text, detecting faces, scanning barcodes, labeling images and recognizing landmarks. You simply pass in data to the ML Kit library and it will give you the information you need - all in a few lines of code.



**Figure 3.5 MLkit features**

If ML Kit's APIs don't cover your use cases, you can always bring your own existing TensorFlow Lite models (Deep learning). Just upload your model on to Firebase, and we'll take care of hosting and serving it to your app. ML Kit acts as an API layer to your custom model, making it easy to run and use. You can also train a custom model using AutoML Vision and then easily deploy that model via ML Kit.

ML Kit gives you both on-device and Cloud APIs, all in a common and simple interface, allowing you to choose the ones that fit your requirements best. The on-device APIs process data quickly and will work even when there's no network connection, while the cloud-based APIs leverage the power of Google Cloud Platform's machine learning technology to give a higher level of accuracy.

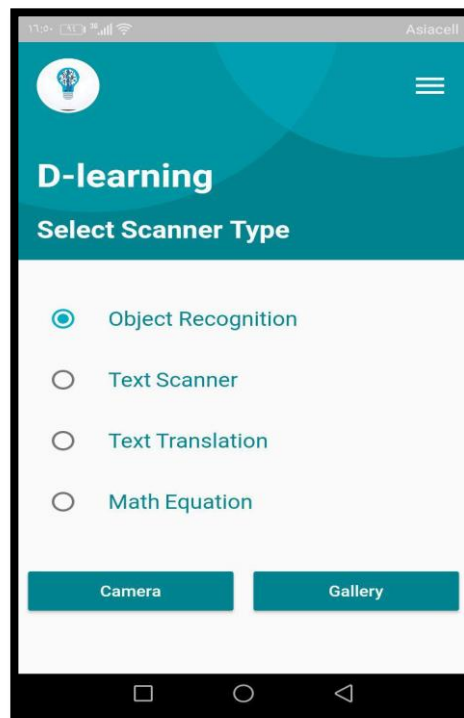


**Figure 3.6 MLkit APIs**

## CHAPTER 4: RESULT

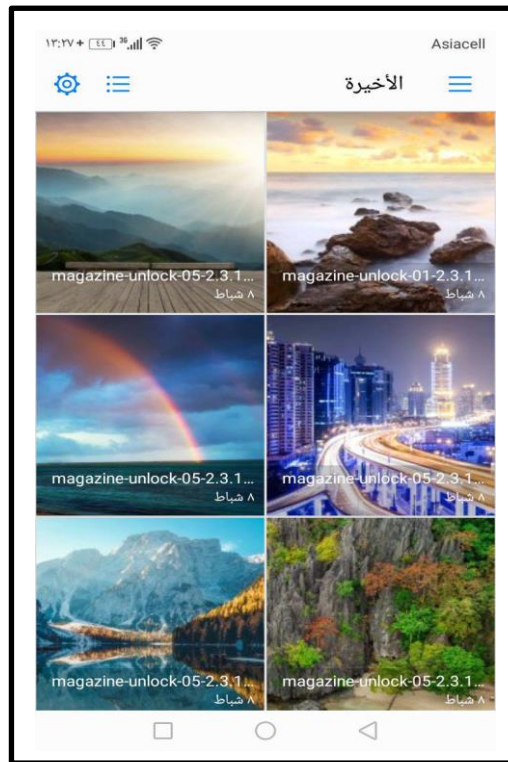
### 4.1 Implementation of ORMA

ORMA uses flutter framework for user interface design, dart language for programming to call deep learning API which is programmed in backend side (Google cloud ) using python programming language to recognize optical character recognition , object recognition and solve math equations. In first interface, the user has ability to choose OCR, text translations, object recognition or solve math equation as shown in figure 4.1



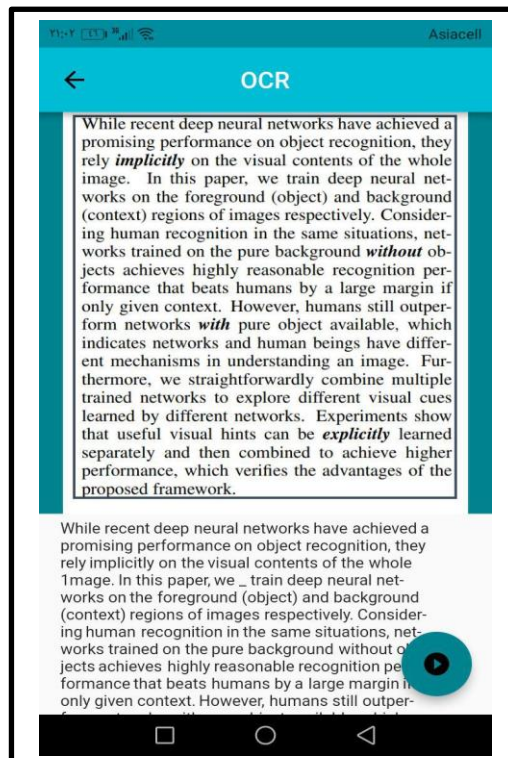
**Figure 4.1: First Interface of ORMA**

In second interface, the user chose image from gallery or take image directly from camera as shown in figure 4.2.



**Figure 4.2: Interface Select Image**

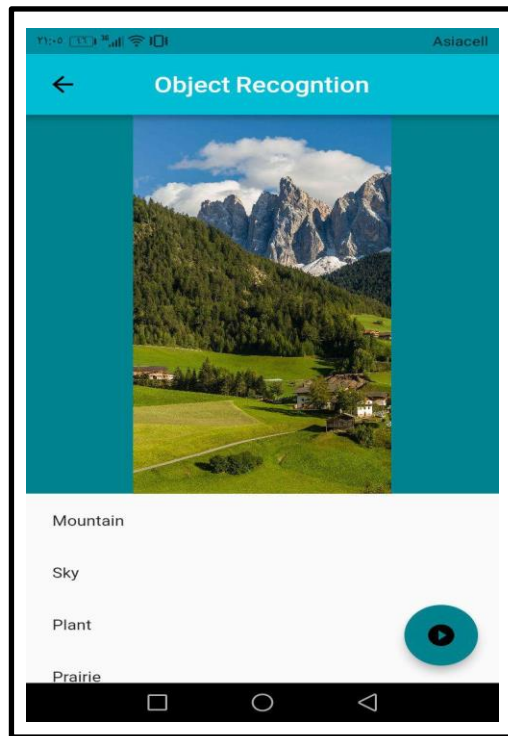
In third interface, display the output of (OCR) as shown in figure 4.3



**Figure 4.3 Text recognition**

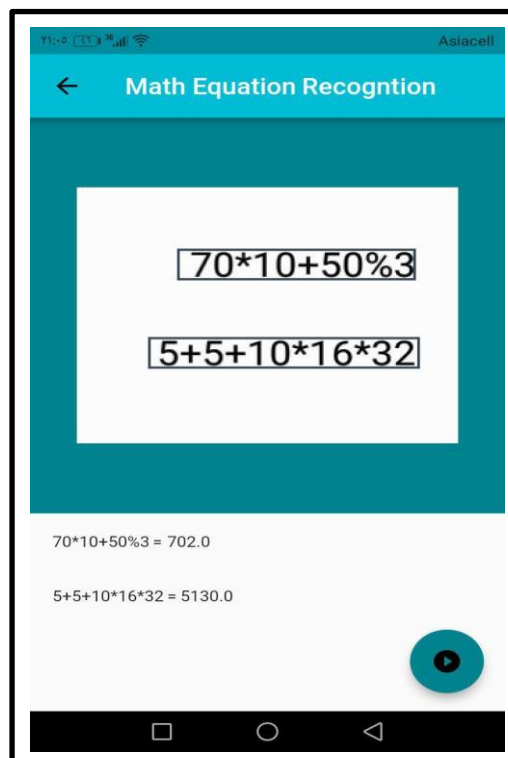
In fourth interface, display the output of object recognition as shown in figure 4.4.





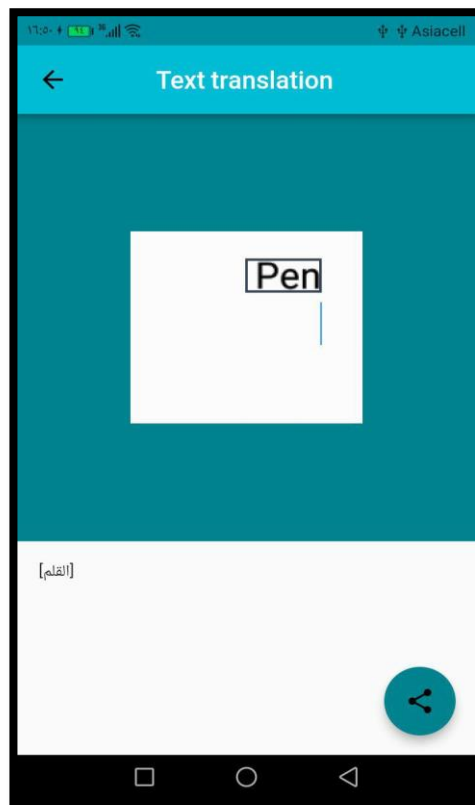
**Figure 4.4: Object recognition**

In fifth interface, display the output of solve mathematical equations as shown in figure 4.5



**Figure 4.5: Solve math equation**

In sixth interface, display the output of text translation as shown in figure 4.6



**Figure 4.6: Text translation**

In all interface we have (shared) button if we click it will enable as to save result in pdf format or print it as shown in figure 4.7 and figure 4.8.

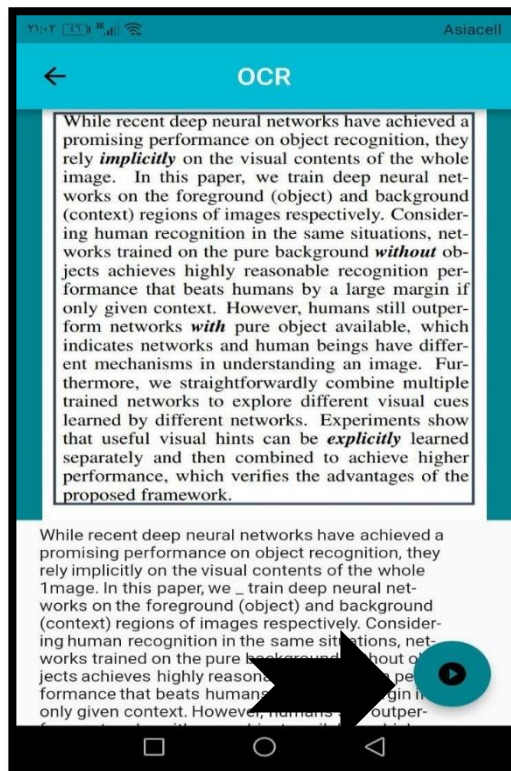


Figure 4.7: shared button

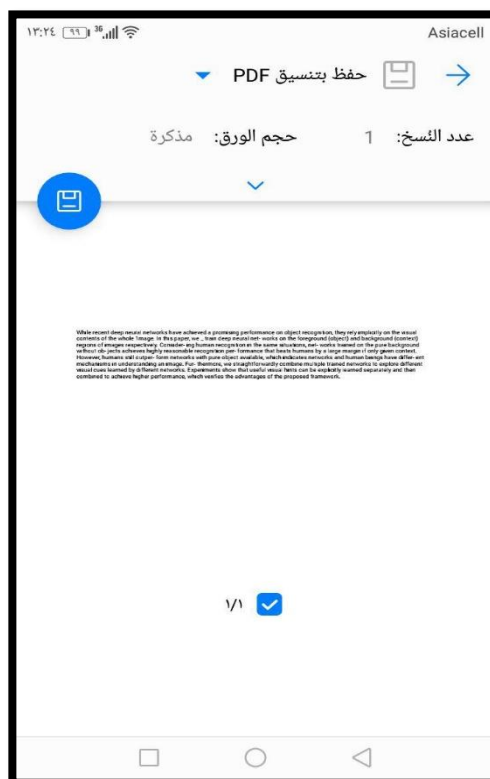


Figure 4.8: Share interface

## CHAPTER 5: THE CONCLUSIONS AND RECOMMENDATIONS

### 5.1 Conclusions

In our project we made application with ability to recognized more than 3000 objects in approximate accuracy of 95% and we exceed the classical model limitations for object recognitions that recognized limit number of object, the app provide the confidence of each object recognized, also they have ability to recognized text in English language in approximate accuracy of 92% , the applications can also detection the mathematical equations and solve it, not only one equation in image, but also more than one equation in image, and the application can translate any text in image to Arabic language with approximate accuracy of 87%, also the application can shared result as pdf and word file that we have ability to modify it and also we can print result

### 5.2 Recommendations

The future work of our app:

- Add sound recognition.
- ability to solve complex math equations

## REFERENCES

- [1] S. Kaur, S. Saini, and R. Vashisht, "Mobile Computing," *Advance in Electronic and Electric Engineering*, vol. 3, no. 6. pp. 399–408, 2013.
- [2] G. Deepak and B. Pradeep, "Challenging Issues and Limitations of Mobile Computing," *Int. J. Computer Technology & ...*, vol. 3, no. 1. pp. 177–181, 2012.
- [3] K. A. More and M. P. Chandran, "Native Vs Hybrid Apps," *Int. J. Curr. Trends Eng. Res.*, vol. 2, no. 6, pp. 563–572, 2016.
- [4] Smith, C., McGuire, B., Huang, T., & Yang, G. (2006, December). History of Artificial Intelligent [Scholarly project]. Retrieved November 20, 2017.
- [5] T. M. Mitchell, Machine learning, McGraw-Hill, New York, 1997.
- [6] M. Richardson, A. Prakash, and E. Brill, Beyond pagerank: machine learning for static ranking, Proceedings of the 15th international conference on World Wide Web, WWW (L. Carr, D. De Roure, A. Iyengar, C.A. Goble, and M. Dahlin, eds.), ACM, 2006, pp. 707–715.
- [7] Yao, M. (2017, August 19). Factories Of The Future Need AI To Survive And Compete. Retrieved November 23, 2017.
- [8] Dian L, Dongmei S. Contactless palmprint recognition based on convolutional neural network. IEEE 13th Int Conf Signal Process. 2016.
- [9] Thoma M. Analysis and Optimization of Convolutional Neural Network Architectures [Internet]. 2017.
- [10] Ahmed Q. AL-Thahab, Image Recognition Using Combination of Multiwavelet and Radon Transforms with Neural Network, *Babylon University, College of Engineering*, 2015

- [11] Laine M. and Olli S. N., “A standalone OCR system for mobile camera-phones”, Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium, pp.1-5, Sept. 2016
- [12] Anwar Hassan Mahdy ,Characters Recognition based on Geometrical Features, engineering journal ,Baghdad 21016
- [13] Lucas Paletta A Mobile Vision System for Urban Detection with Informative, Digital Image Processing 2107
- [14] Román Osorio, N. Bustos, Object Recognition using App Android, IEEE 13th Int Conf Signal Process 2017.
- [15] Wen-Yen Tseng, Kai-Hsiang Chen,An Effective Object Recognition System by a Mobile Application, International Conference on Ubi-media Computing and Workshops,2018
- [16] Yue Liu, Ju Yang, Mingjun Liu,Recognition of QR Code with mobile phones, *IEEE Trans* 2018
- [17] Chucai Yi, , and Yingli Tian, Scene Text Recognition in Mobile Applications by Character Descriptor and Structure Configuration, IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 23, NO. 7, JULY 2018
- [18] Weinman, E. Learned-Miller, and A. R. Hanson, “Scene text recognition using similarity and a lexicon with sparse belief propagation,” *IEEE Trans. Pattern*,2018