Food Recognition on Mobile Phones

Muhammed Mehmet ÜNAL (21400248) ~ Yusuf KARAGÖZ (21501298)

***Abstract*— One of the most important actions we need to keep our lives alive is to eat. We provide the necessary energy for our bodies by eating. We also have fun. Most of us like to eat. Our project will be liked by people who like to eat. The main purpose of our project is to show the name of the food that we take with the camera on the screen. In this way, we can easily learn the names of the things we eat.**

I. INTRODUCTION

Eating is one of the most important activities in human life. We need to eat to continue our daily life. Did we ever think about having information about what we ate? We eat at the restaurant, at school, at work. The main purpose of our project is to reach the names of the food we eat immediately.

Thanks to our project, users will be able to learn the names of the dishes easily and quickly by taking the image of their meals whenever and wherever they want. They will not have to search for the names of the food they are curious about on the internet or on other platforms. All they have to do is show and search them in the official program.

II. RELATED WORK

Different algorithms are used in the food recognition program to automatically identify food. These algorithms consist of the following stages. Image acquisition, image processing, image partitioning, image feature extraction and image classification. The first food recognition algorithm on the mobile device was published in 2009 by Joutou. The following work has been done on food recognition.

- **Joutou and Yanai, 2009:** This system is the first system to use food recognition algorithm in mobile devices. This system was first used in the diagnosis of simple fruit-vegetables. 61.34% success was achieved in 50 different categories. Since 2010, it has been developed to recognize 85 different food categories. In this study, success rate was 62.5%.
- **Puri et al., 2009:** The aim of the study was to estimate food recognition and volume. The system is designed to identify multiple images in a plate. All algorithms for image segmentation and classification are executed on a server. The mobile phone is only used for recording audio and video data. 150 different classifications were used and more than 90% success was achieved.
- **Kong and Tan, 2012:** This article introduces the DietCam mobile phone application. This system is similar to Puri et al system for food detection and volume estimation. The most important difference in this study is that both systems are executed only on the basis of image. Optical character recognition technique is used in the system to read the food names. 92% success was achieved.
- **Rahman et al., 2012:** In this study, Pickering and Kerr have proposed a new texture feature that will improve the accuracy of food recognition on the mobile platform. This feature is based on Gabor filter. The main difference from other studies is that it uses multi-scale and orientation images for foodstuffs. 95% success was achieved in 209 categories.
- **Kawano and Yanai, 2013:** In this study, it is suggested that all operations for food recognition process take place on the mobile phone. This is the first application that accomplishes this goal. This system aims at real-time food recognition. It is aimed to be in harmony with the user. The system is optimized using Fisher vectors and different image features. Perceived foods are shown in boxes. Better recognition processing time has been reached.(From 0.26 seconds to 0.065 seconds) The number of categories was doubled (from 50 to 100). In 2015 this system was implemented on the android platform for public use. 81.55% success was achieved in 50 different classifications and 79.2% success was achieved in 100 different classifications.
- **Anthimopoulos et al., 2013:** This system has been developed to help patients with carbohydrate count and diabetes. Six different categories are used in image classification (meat, breaded food,

rice, pasta, potatoes and vegetables). In this system, different foods from these 6 categories were not used. 87% success was achieved in 6 different classifications.

- **Pouladzadeh, Shirmohammadi and Arici, 2013:** In this system, automatic food measuring system is aimed. It is to estimate the calories eaten before and after the food as well as the nutrient detection. 92.21% success was achieved in 30 different classifications.[1]

  More information can be found in [1].

The method we use in our study is as follows. Image processing technique is performed on the desired image and compared with the images in the data set using the deep learning method. The name of the matching image in the data set is displayed.

III. Method

We realized our project based on deep learning logic. We first worked on *TensorFlow*, the official popular deep learning library that we identified, and matched it with the appropriate data set. Then we showed the food name. We realized our project by using *keras* library and tensorflow platform on anaconda which has python distribution. To do this, on anaconda tool will be installed primarily the *Tensorflow* and then the *keras* library will be installed.The working principle is shown in Figure 1.

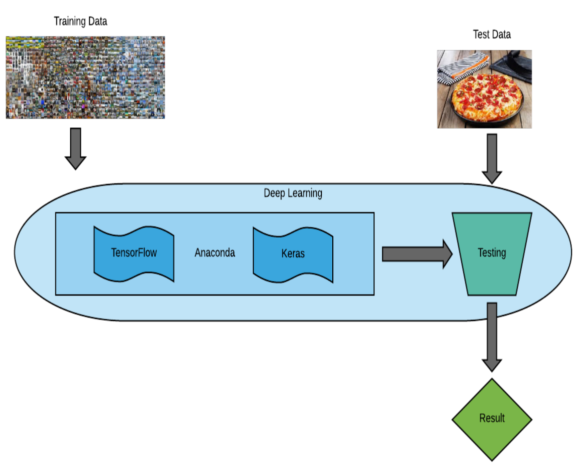

Fig. 1. Overview of the algorithm

Then we started to import our project on android. First, we carried our program to the internet using Heroku environment. Then we created the interface of our program in android studio environment. Finally, we uploaded the picture we received in our application to Heroku environment and showed the answer in the application by pulling.

A. *Deep Learning*

We can define deep learning as systems that can learn and extract based on data. They are capable of detecting and interpreting very high amount of data. The deep learning system has multiple layers and a separate operation is performed on each layer. As shown in figure 2; with the deep learning, in the image classification process, the pixels,the motifs by edge combination formed, these motifs combine to form the objects by combining the pieces of the object and the pieces of the object.

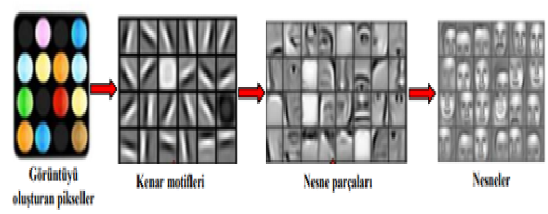

Fig. 2. Image classification with deep learning

Deep learning consists of different layers. The most important of these is the convolution layer. This layer is known as the conversion layer.As shown in figure 3; this conversion process is based on the process of circulating a particular filter over the entire image. Specified small size matrices move over the entire image matrix to highlight the attributes in the image. And a new image matrix is obtained in the image size. Filters form the output data by applying the convolution process to the images from the previous layer.

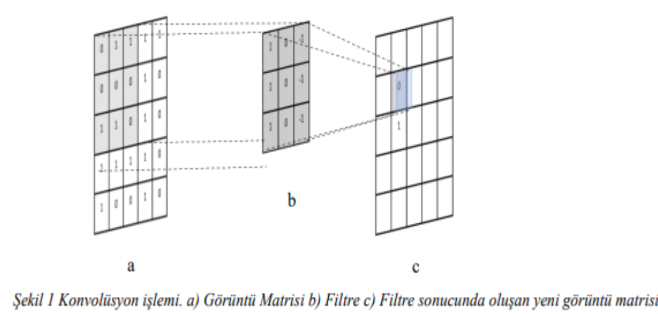

Fig. 3. Convolution process, a) image matrix b) filter c)new image matrix as a result of a filter

As a result of the convolution process, the characteristics of each filter are discovered. Determines which

regions of the data are important. And the process is realized by concentrating on these regions.
Deep learning can do the following. Can detect and identify objects. Interpret the pictures. Makes inferences about the pictures and makes predictions. We have trained the network with the data set in our project and we have provided the correct matching of the food pictures we have shown.

B. Anaconda Navigator

Anaconda is python distribution for those who want to use python for data science and similar scientific applications. Data science includes libraries that are frequently used in topics such as artificial intelligence. When you set up anaconda python will also be set up[1].

C. Keras Library

Keras is a high-level API to build and train deep learning models. It's used for fast prototyping, advanced research, and production. *Keras* has a simple, consistent interface optimized for common use cases. It provides clear and actionable feedback for user errors. It supports both common and repeating networks and combinations of the two. It runs smoothly on CPU and GPU[2].

D. TensorFlow

TensorFlow is an open-source machine learning library for research and production. *TensorFlow* offers APIs for beginners and experts to develop for desktop, mobile, web, and cloud. Originally developed by researchers and engineers from the Google Brain team. We will be using transfer learning, which means we are starting with a model that has been already trained on another problem. We will then retrain it on a similar problem. Deep learning from scratch can take days, but transfer learning can be done in short order[3].
Before you start any training, you'll need a set of images to teach the model about the new classes you want to recognize. We received the data needed to train the model at ' http://foodcam.mobi/dataset100.html ' .
Then the training of the model begins. The command line is executed on the Anaconda and the following commands are written.

```
>>> python –m retrain\
>>> —bottleneck_dif=tf_files/bottlenecks\
>>>—how_many_training_steps=200\
>>>—model_dir=tf_files/models/\
>>>—summaries_dir=tf_files/training_summaries
    mobilenet_0.50_224\
>>>—output_graph=tf_files/retrained_graph.pb\
>>>—output_labels=tf_files/retrained_labels.
    txt\
>>>—architecture=mobilenet_0.50_224\
>>>image_dir=tf_files/dataset
```

This training phase takes some time. This script downloads the pre-trained model, adds a new final layer, and trains that layer on the food photos you've downloaded. You can specify how many steps the training will take with this command.

```
>>>—how_many_training_steps=200\
```

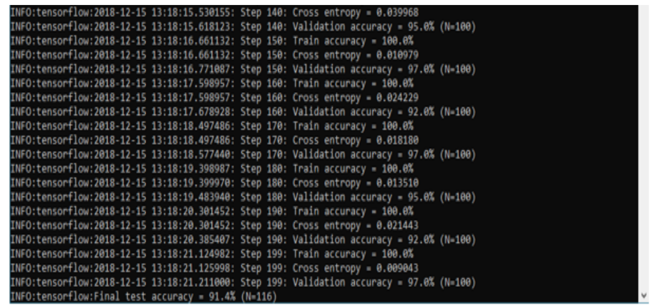

Fig. 4. Image of the system at run

When the system starts up, the view is as in figure 4.

- The training accuracy shows the percentage of the images used in the current training batch that were labeled with the correct class.

- Validation accuracy: The validation accuracy is the precision (percentage of correctly-labelled images) on a randomly-selected group of images from a different set.

- Cross entropy is a loss function that gives a glimpse into how well the learning process is progressing. (Lower numbers are better.)

- Final test accuracy shows the accuracy percentage after the end of training. 91.4% success was achieved in our sample application.

E. Testing

In our study we used different food groups. There are 11 different food classes in this phase of our project. There are also differences in the same food groups. For example, rice pilaf and bulgur pilaf are below the rice category. In addition, the system can distinguish different apple colors. Apple class; red, green and yellow apple.
After the completion of the training, we give our sample picture to the project for matching.

```
>>> python -m label_image\
>>>--graph=tf_files/retrained_graph.pb\
>>>--image=../../pizza.jpg
```

With this command, we have included our picture in the project.
The retraining script writes data to the following file: tf_files/retrained_graph.pb
The result is as follows for the pizza picture.
As shown in figure 5; our program says that the picture is 99% pizza.

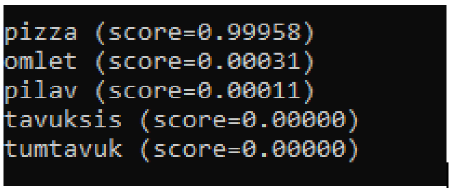

Fig. 5. Output image

In Figure 6, rice pilaf was included in the system and 81% match was achieved.

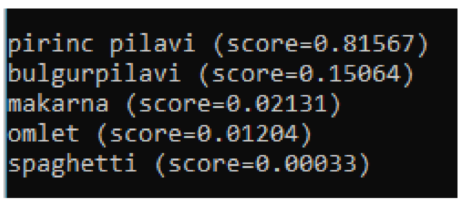

Fig. 6. Output image

In Figure 7, Red apples are separated from other apple groups by color and 99% match is achieved.

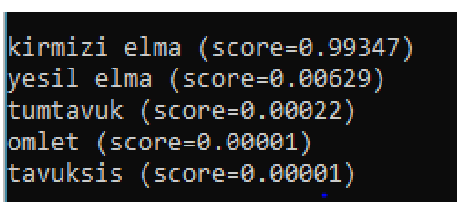

Fig. 7. Output image

F. Heroku

Heroku is a cloud computing application infrastructure service provider. It enables us to move our applications to the internet in languages such as Java, php, python without the need of technical server knowledge.

First, we installed our project with github and then shared it with Heroku. We opened our project in heroku environment and created the interface as in Figure 8.

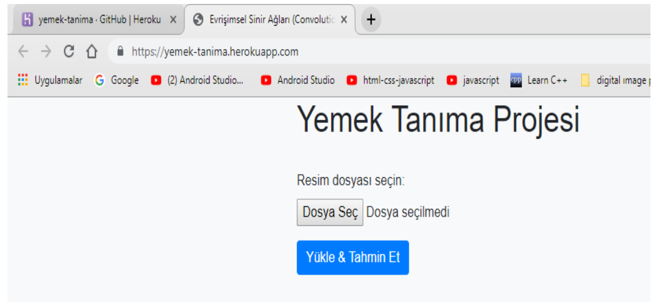

Fig. 8. Interface created with Heroku

We received the result of our uploaded image as in Figure 9.

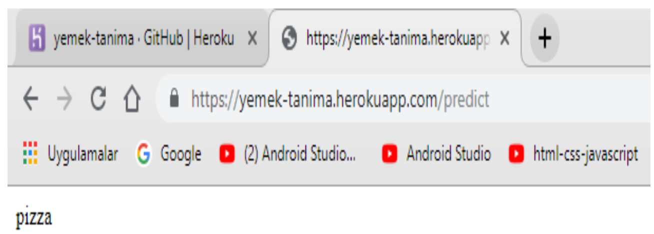

Fig. 9. Return of result

Then we sent pictures to heroku platform with android studio and showed the result to the user by pulling from the internet.

G. Android Studio

We realized our project on android platform for easy use at every moment of the day. We take the picture from the gallery or take it with the camera and upload it to the machine learning environment. In short, we upload data to the internet. Then we show the result of the forecast to the user by pulling from the internet.
We used the OkHttp library in our project. This library is used to send and receive http requests to the web server. We shared data and media in this way. The reason we use OkHttp is that it is advantageous. For example; fast load and bandwidth savings. Reduces delay time. Eliminates common connection problems.

H. Use Of Our Project and Interface

As shown in Figure 10, we select a picture by clicking on the select button first in our opened project. If we want, we can upload pictures by camera or by selecting from the gallery. Then we click the upload button and upload our picture to our project in heroku environment. Finally, we show our prediction result to the user.

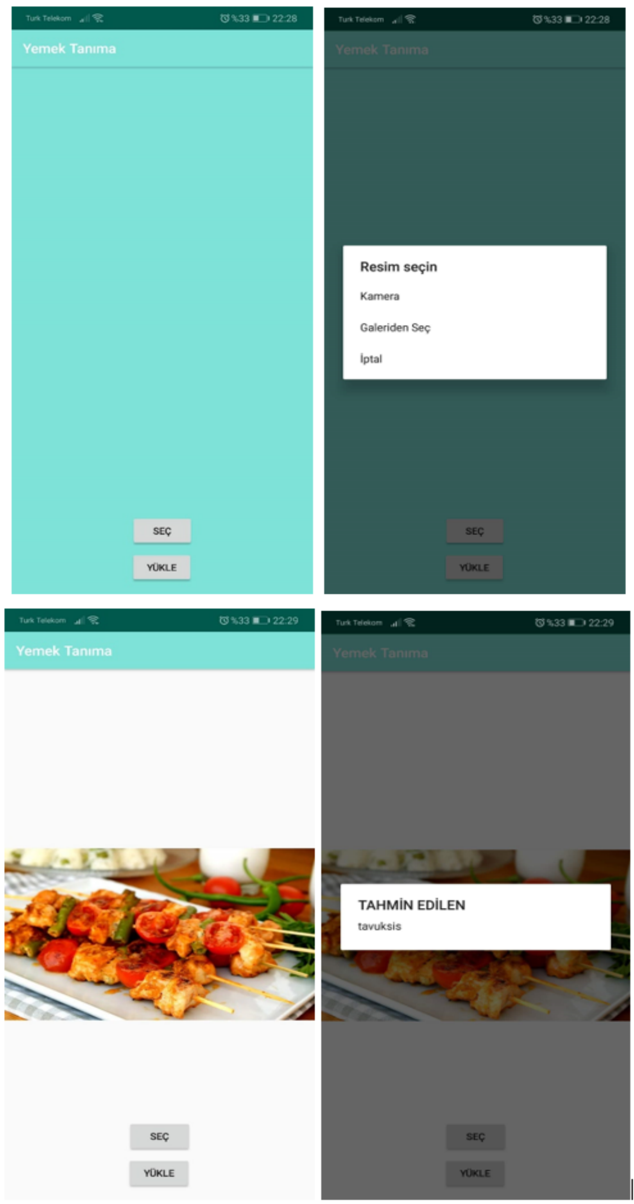

Fig. 10. Case study of our project

IV. RESULT/FINDINGS

We realized our project on anaconda using *TensorFlow* and *keras* libraries. First of all, we have trained our network with our data set and we have given the image we set for testing as input into the system.
As the next step, we uploaded our project to heroku environment. We created an interface with Android studio and uploaded our picture to the system. Lastly, we showed the user the prediction result.
As a result of various filtering and subdivision for each frame on the image, the objects detected on the picture frame are classified by inserting them into the pre-trained network.
We finalized each dish according to its specific characteristics as separate groupings.
The result was successful at 99% and we learned the name of the given dish correctly.

V. SUMMARY

In this project we tried to learn the names of the food by taking the image. Using the *TensorFlow* library on anaconda, we have trained the system with the meal data set. Then we uploaded our work to Heroku environment. We created our project interface with Android Studio. We uploaded the picture to the internet and uploaded it to the trained system and showed the user the prediction result. We've provided the pairing. We have achieved 99% success.

REFERENCES

[1] https://goo.gl/oCBckE
[2] http://matkafasi.com/112536/anaconda-nedir-nasil-kurulur
[3] https://www.tensorflow.org/guide/keras
[4] https://codelabs.developers.google.com/codelabs/tensorflow-for-poets/#()