Prototype mobile application definitions fresh products based on neural network

L A Korobova¹, I A Matytsina², I S Tolstova¹ and M S Mironova³

¹Department of Higher Mathematics and Information Technologies, Voronezh State University of Engineering Technologies, 19 Revolution Avenue, Voronezh, 394036, Russia
²Faculty of Secondary Professional Education, Voronezh State University of Engineering Technologies, Leninsky Prospect 14, Voronezh, 394029, Russia
³Department of Mathematics, Air Force Military Educational and Scientific Center «Air Force Academy named after Professor N.E. Zhukovsky and Y.A. Gagarin», 54 A, street of Old Bolsheviks, 394064

E-mail: irina210390@mail.ru

Abstract. The article discusses the functioning of the image classifier. The results of the process of research and implementation of the food freshness classifier using mobile phones to obtain product images are presented. Comparison of other approaches to reducing the risk of food poisoning is given. Provides research information on the collection of images for the classification of food and the classification of their freshness. A working android application for recognizing the freshness of food has been implemented. Based on the analysis of the results of the training sample, the objects under study are ranked in order to determine the freshness of the product.

1. Introduction

Freshness is a broad and comprehensive concept. It is inextricably linked with the quality of products [1]. The quality of food is determined in the laboratory using the methods of chemical and microbiological analysis. These types of analysis are expensive expertise and require the use of specialized equipment and a variety of reagents. The entire examination process takes a long time, from several hours to several days. There is no single algorithm for the examination. Each product is a biological system with its own structure. And as in any system, the interaction and connections of the components of the product within themselves and in relation to the environment are considered here. The examination process is subject to regulatory documents and is regulated by the state. Everything is officially recorded by a protocol or an examination certificate in special quality control bodies.

One of the main priorities for providing the population with food is the quality of food. With the advancement of information technology, many tools have emerged to reduce the risk of food poisoning. However, not all of them are intended for mass use and are not always useful. This paper proposes a toolkit for recognizing the freshness of products purchased not in specialized outlets, but, for example, in the markets, having only an image of the product. A special feature of the toolkit is that it becomes possible to determine the freshness of food products "on the spot", having only a mobile phone with a camera. At the
first stage, the product image is scanned (photographed). Further, the application for the mobile device (phone) will determine which product is in the image transmitted from the mobile phone camera, and whether this product is fresh. The information is then displayed on the user's screen. In addition to recognition, the application has help information on how to recognize the freshness of a product in other ways, for example, by smell. The most common target for mass adoption of an application is a mobile phone.

From the above, it follows that the subject of the article is relevant and, indeed, there is a need to develop a mobile application to determine the freshness of the product.

To achieve this goal, it is necessary to solve the following tasks:
- consider technologies for constructing classifiers;
- collect the necessary data to create classifiers;
- create a classifier of products and their freshness;
- to develop an application using the created classifiers to reduce the risk of food poisoning;
- analyze the results.

2. Analysis of analogs of software applications

After analyzing the market for applications designed to reduce the likelihood of getting food poisoning, the following categories of applications can be distinguished:

1) applications that store information about the date of manufacture based on the entered product data;
2) applications that issue information about a product by a barcode;
3) applications for finding rotten places on a product based on its photo.

Applications of the first category are designed to keep track of purchased products and track their expiration date. They provide a user-friendly interface that allows you to: keep track of the list of purchased products and products to be purchased (pre-planned), indication of the freshness of the product, expressed both graphically and numerically, notification if the product expires, and many other useful features to prevent food poisoning. The disadvantage of this approach is the inability to determine the freshness of a product by any indication, if the product is not purchased at a specialized outlet that does not provide official information from the manufacturer. Due to the fact that such applications do not require time to train the neural network, the development time for this application is minimal. Examples of applications in this category are: "Shopping list - My List", "Purchase history: QR check scanner, expense tracking", "Shopping list - Buy a Baton!" [2 - 5].

Often, applications in the first category include the functionality of applications in the second category. These applications allow you to determine the expiration date of the product, its name and composition by the barcode. Data for training models to recognize product barcodes is widespread and readily available in the public domain, thus reducing application implementation time. The disadvantage of this approach is the same as for the first category of applications. Examples of applications in this category are: "Product Accounting, Expiry dates, Purchase List", "Expiration date", "BEEP - barcode and expiration date scanner" [6, 7].

Applications of the third category are not widely used due to the complexity of their implementation. The network does not have the necessary data to train the neural network, so the data must be collected from scratch. In this regard, the time for developing an application will take much longer than for applications of the first and second categories. However, the advantage of this category is the ability to recognize the freshness of products purchased not in specialized retail outlets and obtained in any other way, having only an image of the product. Examples of applications in this category are: "Fresh fruit detector - quality control" [8].

3. Choosing a solution method
A modern mobile device has a camera, so the classification of images was chosen for recognition [9]. To create a system that can separate one class of images from others, a neural network has been developed and tuned [10]. The conducted research and analysis of the image quality allowed us to conclude that for training a neural network, it is necessary to collect images that meet the following requirements:

- it is necessary to collect images of products without taking into account the religious, national and cultural characteristics of people, as well as time periods (wartime or cat-ememas);
- it is necessary to collect such images to determine the freshness of products, which depict products that do not cause poisoning to the body of a healthy person;
- it is necessary to collect images of any products that do not cause poisoning for a healthy person, indicating the number of grams for each, they will also be considered fresh and edible.

The presented development allows you to provide assistance in determining the freshness of food. Bread, tomatoes and beef are selected as examples. This development is useful for buyers who make purchases in the markets and in unauthorized retail outlets. The main requirement is that buyers have a mobile phone with a camera. Also, this application will be useful in enterprises where quality control of products is required.

Functioning of the image classifier.

Neural networks are used to build an image classifier. To optimize the work of the classifier, it is necessary to create convolutional neural networks. The structure of a neural network in programming interprets a biological system. Therefore, the operation of the network is similar to the calculations taking place in the brain of an animal. Thanks to this structure, the neural network can analyze, memorize various information and reproduce it from its memory.

A convolutional neural network is used to classify images. In contrast to the usual perceptron, convolutional networks are more capable of recognizing patterns in images. They are based on filters (kernels) that recognize certain characteristics of the image, for example, straight lines. Kernel is a numerical matrix that "learns" to find some characteristics in images. The filter floats along the image and says if there is any characteristic we are looking for in a particular part of it. To obtain a similar answer, the convolution operation is used (figure 1), which is the sum of the products of the filter elements and the matrix of input signals [11].

Description of the work of the Tensorflow framework

Today, there is no need to create your own library that allows you to build simple neural network models. Instead of creating a library from scratch, you can use frameworks and libraries to build a high-level neural network. One of these frameworks is Tensorflow.

Tensorflow allows you to create and train models using the high-level Keras API, making it easy to get started with Tensorflow and machine learning [10].

In order not to build a neural network architecture from scratch, a ready-made network called MobileNetV2 was used.

Development of classifiers

To retrain the models, a ready-made MobileNetV2 model was used, which was retrained on the collected data.

You need to get scripts from Tensorflow:
- retrain.py for training the model [9];
- install the tflite_converter module for python for converting to a tflite model so that it can be used in android and ios applications;
- tflite_tester.py for testing the model on images [10].

To train the product recognition model, you need to place the selected images in the train_images directory, grouping the image classes by directories. The directories themselves must have the names of the image classes they contain. Images for testing will be in the test_images directory without grouping. The
retrain.py and tflite_converter.py scripts must be at the same level as the test_images and train_images directories.

The directory structure for retraining the model for food recognition contains the bread, beef, tomato, another directories. The bread catalog contains various images of bread. The beef directory contains images of beef. The tomato directory contains images of tomatoes. In another it is necessary to arrange a variety of objects that do not contain food. This must be done to exclude non-food products from other objects. For example, images of the hands of various people are placed here so that the hands do not influence the definition of the product. The structure of directories and files is shown in figure 1.

![Figure 1. File structure for retraining for food recognition.](image)

There are two ways to train the model to determine the freshness of foods. The first way is to split each catalog from the product recognition training model into 2 subdirectories fresh (fresh) and spoiled (not fresh or spoiled) by prefixing the product name. For example, the bread directory will be split into a bread_fresh directory, which will contain images of fresh bread, and a bread_spoiled directory, which will contain photographs of spoiled bread.

The second way is to combine all fresh food into the fresh directory, and all the spoiled ones into the spoiled directory. This method was chosen for the implementation of the application, since it can be implemented faster.

The directory structure for retraining a model to recognize food freshness is very similar to the structure for defining food itself. Only the class catalogs and the images in them will be different. Consists of fresh, spoiled and another directories. Fresh needs to include images of fresh food products. Spoiled will contain photographs with spoiled food. Another catalog is analogous to the catalog in the product definition model. The hierarchy of files and directories is shown in figure 2.
The process of obtaining a finished model for both recognizers is similar [12]. Therefore, further training and testing will be described only for the food recognizer.

After receiving the tflite model, it can be tested on images. For this, the tflite_tester.py script is used, which is launched by the tflite_tester.py python language command. An example of testing is shown in figure 3 (tested image) and figure 4 (the result of solving the neural network and the label).

To understand what the data from the array means, you need to refer to the file with labels (see figure 5), which was generated when training the model.

The definition of the neural network result is constructed as follows. From the resulting array of values

```
[[1.2624279e-05 3.3685043e-05 9.9995375e-01 2.0395956e-10]]
```

we determine the ordinal number of the
largest, which is 9.9995375e-01 (when rounded to hundredths 0.99 or 99%). From the text file with labels, select the label by the ordinal value, which is "bread", which means "bread".

Based on the data received, the network considers with a 99% probability that the image contains bread. The same operations must be done to train the model to determine the freshness of the product.

As a result of retraining the model for recognizing food products and recognizing their freshness, the following was obtained:

- 2 files with retrained models of tflite format;
- 2 files with tags (enumeration of classes) txt format.

The obtained data can be used to develop an application using the Tensorflow libraries. The most popular neural network library for mobile phones is Tensorflow Lite.

4. Results
We will demonstrate an example of how the application works by recognizing the freshness of white bread. After launching the application, the main menu activity is displayed first, shown in figure 6. The main menu consists of a list with clickable items from which you can go to the following activities: freshness recognizer, list of products, list of instructions. The last item "Exit" is the exit from the application.

![Figure 6. Main menu screen.](image)

![Figure 7. Product list screen.](image)

When you click on the item "Products", a list of products opens, shown in figure 7. This list provides products about which the developed application has information. Each element of the list is clickable and goes to a screen with information on determining the freshness of the selected product, its visual characteristics, the smell of the product and other external features.

To switch to the freshness recognizer, select the "Freshness recognizer" item from the main menu. When the screen is opened, permission is requested to use cameras, if it is absent. After approval, the product is captured by the camera screen of the mobile device, and the camera image is captured. Then the image is processed using a product definition classifier and a product freshness classifier.

The results are displayed in the application window. The first line displays the name of the product and the probability with which the network believes that this is the same product. The second line displays the following information: fresh product or not and with what probability the recognition occurred. If the product is not defined, then it is displayed that the product is unknown. If a product is detected, a screen with detailed information about the recognized product will be displayed.

Depending on whether the product is unknown, fresh or spoiled, the visualization window of the mobile application changes color:

- yellow - the product is unknown;
- red - the product is spoiled;
- green - the product is fresh.

The last menu item in figure 6 is instructions. When you click on it, the application goes to a screen with a list of instructions for working with the application. The main ones are: - how to use the product freshness recognizer; - how to use the data for determining the freshness of the product: - about the application. Each element of the list is clickable, and when you click on the corresponding item, you go to the menu of the "Instructions" screen for working with the application.

5. The discussion of the results
The paper deals with the issues of determining the quality of food. Based on this, classifiers have been developed to determine the type of product and the likelihood of spoilage of this product. The essence of the problem is the construction of image classifiers capable of recognizing the "poison" in a food product. To solve it, the Tensorflow framework is used. It allows you to retrain ready-made classification models at the top level, which contributes to faster learning.

Depending on the length of the video coming from the mobile device, a corresponding number of images are obtained: if the video is 40-50 seconds long, ffmpeg creates about 900-1000 images. Along with these images, Google images have been added to have a wider range. The developed proto-type allows you to determine the freshness of the product thanks to the phone camera and the application installed on it. Experimental studies have shown that the accuracy of the application is 81%. During the research, 1000 computational experiments were carried out. Of these, the software application performed correctly in 810 experiments, and falsely in 190. The result of the program is shown in figure 8 - a fragment of the mobile phone screen. The phone screen displays text information. It can be seen that the test product was recognized as bread with a probability of 65.39% and that the product was spoiled with a probability of -75.5%. The color on the phone screen, red, confirms that the bread is spoiled.

Figure 8. Experiment result - determination of fresh bread.
Since there are no analogues in the field of mobile applications for determining the freshness of a product by its image, our development is relevant. But it is necessary to continue scientific research in the field of teaching and bring the recognition accuracy to 100%.

References

[1] Derkanosova N M, Korobova L A and Ukhina E Yu 2016 Design and quality assurance of food products (for example, bakery products) monograph (Voronezh in Russian)

[2] [Electronic resource] - Application "Shopping List - My List" - https://play.google.com/store/apps/details?id=com.simplesmartsoft.mylist

[3] [Electronic resource] - Application "Purchase history: QR check scanner, expense tracking" - https://play.google.com/store/apps/details?id=matrix.rparse

[4] [Electronic resource] - Application "Shopping List - Buy a Baton!" - https://play.google.com/store/apps/details?id=com.buymeapie.bmp

[5] [Electronic resource] - Application "Product Accounting, Expiry dates, Purchase List" - https://play.google.com/store/apps/details?id=com.chestersw.foodlist

[6] [Electronic resource] - Application "Expiry date" - https://play.google.com/store/apps/details?id=info.shelflife.shelllife

[7] [Electronic resource] - Application "BEEP - barcode scanner and expiration date" - https://play.google.com/store/apps/details?id=com.bgpworks.beep

[8] [Electronic resource] - Application "Fresh fruit detector - quality check" - https://play.google.com/store/apps/details?id=com.fruit.detector.app.vegetable.freshness.detection

[9] [Electronic resource] - What is image classification - http://desktop.arcgis.com/ru/arcmap/10.3/guide-books/extensions/spatial-analyst/image-classification/what-is-image-classification.htm (date accessed 07.09.2019).

[10] Ryabchunov D V and Korobova L A 2020 Review of the structure and operation of neural networks Modeling of energy-information processes. VIII National scientific and practical conference with international participation (Voronezh) pp 572-581 (in Russian)

[11] Ryabchunov D V and Korobova L A 2019 Image classification technology in recognizing the freshness of products on the example of beef Food security: scientific, personnel and information support. Collection of scientific articles and reports of the VI International scientific and practical conference (Voronezh) pp. 596-600 (in Russia)

[12] Abramov G, Korobova L and Matytsina I 2020 Development of algorithm for analysis of sound fragments in medical information systems Journal of Physics: Conference Series. Applied Mathematics, Computational Science and Mechanics: Current Problems S. 012095.