applied sciences

MDPI

Article

Recognition of Food Ingredients—Dataset Analysis

João Louro [1,*], Filipe Fidalgo [1,2] and Ângela Oliveira [1,2]

[1] Polytechnic Institute of Castelo Branco, 6000-767 Castelo Branco, Portugal; ffidalgo@ipcb.pt (F.F.); angelaoliveira@ipcb.pt (Â.O.)
[2] CISeD—Research Centre in Digital Services, 3504-510 Viseu, Portugal
* Correspondence: joao.d.louro@gmail.com

Abstract: Nowadays, food waste is seen as a complex problem with effects on the social, economic, and environmental domains. Even though this view is widely held, it is frequently believed that individual acts have little to no impact on the issue. But just like with recycling, there may be a significant impact if people start adopting more sustainable eating habits. We suggest using a cutting-edge convolutional neural network (CNN) model to identify food in light of these factors. To improve performance, this model makes use of several strategies, such as fine-tuning and transfer learning. Additionally, we suggest using the Selenium library to create a dataset by employing the web scraping technique. This strategy solves the problem that many open-source datasets have with the overrepresentation of foods from the Asian continent by enabling the addition of foods to the dataset in a customized way. First, using the PRISMA methodology, a thorough examination of recent research in this field will be carried out. We will talk about the shortcomings of the most widely used dataset (Food-101), which prevent the ResNet-50 model from performing well. Using this information, a smartphone app that can identify food and suggest recipes based on the ingredients it finds could be developed. This would prevent food waste that results from the lack of imagination and patience of most people. The food recognition model used was the ResNet-50 convolutional neural network, which achieved 90% accuracy for the validation set and roughly 97% accuracy in training.

Keywords: food ingredient recognition; artificial neural network; dataset; web scraping; recommendation system; ResNet-50

Citation: Louro, J.; Fidalgo, F.; Oliveira, Â. Recognition of Food Ingredients—Dataset Analysis. *Appl. Sci.* **2024**, *14*, 5448. https://doi.org/10.3390/app14135448

Academic Editors: Attilio Matera and Francesco Genovese

Received: 15 April 2024
Revised: 17 June 2024
Accepted: 19 June 2024
Published: 23 June 2024

1. Introduction

Recommendation systems have gained significant importance in our daily lives in the era of digital technology. Prominent corporations like Spotify and Netflix utilize diverse tactics to incentivize users to engage with their offerings for extended durations, capitalizing on the data stored in their databases [1].

In the culinary sector, numerous web solutions utilize user-inputted ingredients to recommend recipes. Moreover, besides the absence of real-time food recognition through artificial intelligence, they frequently propose recipes containing ingredients that may not align with the dietary restrictions of individual users. This is demonstrated by the most widely used recipe-recommendation applications mentioned in [2], which do not take into consideration these limitations. Facilitating user assistance, food identification could be achieved effortlessly by directing the smartphone camera towards the food item, enabling the model to discern the food without necessitating the user to manually input the available ingredients. Subsequently, the recipe-recommendation system could utilize an application programming interface (API) to receive the identified food as an input; there are many in this field that can be used, such as Sponnacular [3] or Edamam [4]. The main goal of the recipe-recommendation system, which relies on food identification, is to aid users in repurposing food, thus minimizing food waste.

To recognize food images, it is necessary to implement deep learning models, such as ResNet, EfficientNet, or MobileNet, using powerful tools widely used in machine learning,

such as TensorFlow [5], Keras [6], or NumPy [7]. Subsequently, it is imperative to utilize a dataset to facilitate the training of the model, enabling it to acquire knowledge about the distinctive features present in the training images and subsequently apply this knowledge to previously unobserved images [8].

Therefore, the ResNet-50 model is suggested for food recognition. This model is utilized on the Food-101 dataset and subsequently on the dataset constructed through web scraping. This technique enables the dataset creators to select personalized food choices and simplifies the manual acquisition of high-quality images [9], which would be challenging, if not almost impossible, within the scope of our problem of collecting food images manually.

Initially, the model overfitted, achieving only 20% accuracy for the validation set and approximately 100% for the training set on the Food-101 dataset. In an attempt to overcome this problem, the structure of the convolutional neural network (CNN) was modified, as were some of the model's hyperparameters. However, even with these changes, the accuracy of the validation set remained suboptimal (~70%) for implementation in a mobile application, demonstrating that the use of this model applied to the Food-101 dataset should not be used for food identification, for reasons that will be detailed in Section 3.3. On the other hand, the same model applied to the new dataset obtained excellent results for both the training set (97%) and the validation set (90%).

By addressing these challenges and using advanced machine learning techniques, our work aims to contribute to the field of food recognition by providing a relatively large dataset of foods frequently used by users, with the aim of minimizing food waste and increasing user convenience.

To facilitate a more comprehensive understanding of the essential phases of the project, we have created a flowchart, shown in Figure 1. This representation identifies the planning, analysis, and development phases of the project.

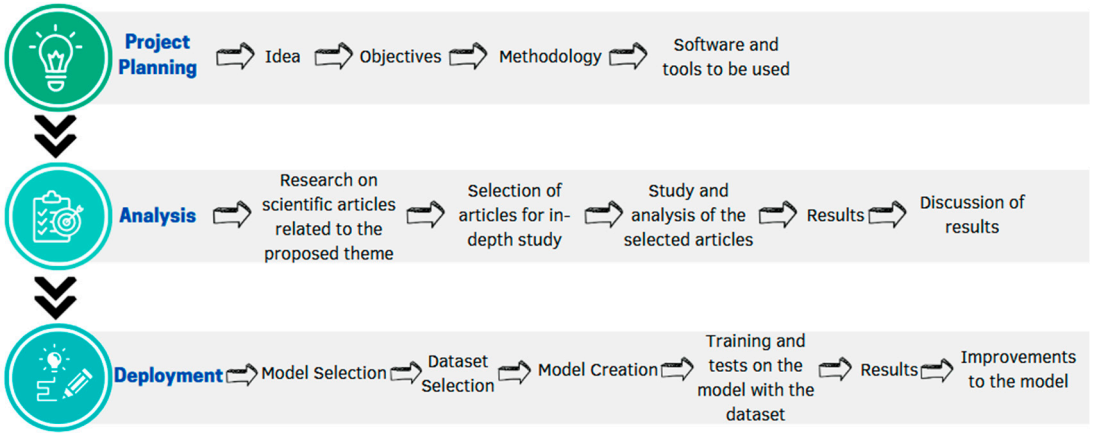

Figure 1. Flowchart of the project phases.

The article adheres to a conventional structure, with Section 2 presenting the research method, which is supported by the systematic review methodology. Section 3 describes the setup used to develop the project, including the proposed machine learning model and the dataset used to train the model, namely Food-101 and the dataset created by us through web scraping. Finally, Section 4 shares our conclusions and suggestions for the implementation of future work.

2. Related Works

The recognition of food and ingredients, as well as the recommendation of recipes, has gained popularity in recent years due to the existence of several algorithms with excellent results in extracting patterns from images. Through an exhaustive search, several articles were found with different approaches to the proposed problem. This systematic review follows the PRISMA methodology (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) [10], which includes the following topics:

A. Research Questions;
B. Inclusion Criteria;
C. Research Strategy;
D. Results;
E. Data Extraction and Analysis;
F. Discussion.

To conduct the research, we formulated research questions on the topics of food image recognition and recipe recommendation. These questions aim to discover solutions that can be used and improved in the future. We therefore came up with the following questions:

- Are there digital solutions capable of recognizing food from images of meals?
- Are there digital solutions that offer recipes based on recognizing ingredients or food images?
- Are there digital solutions that propose meals based on leftovers, taking into account the characteristics of each user?

The inclusion criteria refer to the characteristics that the articles analyzed must have to be selected. The inclusion criteria defined for our study are as follows:

- Criterion 1: Studies carried out between 2015 and 2023;
- Criterion 2: Studies written in English;
- Criterion 3: Studies in which the full text is available;
- Criterion 4: Studies that apply image recognition to cooked food;
- Criterion 5: Studies in which the dataset is available on the web.

A search was conducted using the IEEE Xplore [11], Scopus [12], and ACM Digital Library [13] databases. The search terms we used were 'Dataset' AND 'Image' AND ('Dish arranged' OR 'Ingredient'). The search terms used were "Dataset" AND "Image" AND ("Dish arranged" OR "Ingredient"). The search was carried out between October and November 2023. After applying criterion 1, we identified a total of 114 scientific studies, 39 from IEEEXplore, 63 from Scopus, and 12 from ACM. This is illustrated in Figure 2. We then applied criterion 2 and removed the duplicates, resulting in 106 studies. We conducted a general analysis of these studies, applying criteria 3 and 4. Finally, criterion 5 was applied, resulting in a full-text analysis of the 61 remaining studies. Based on these criteria, the most relevant articles were selected, resulting in 13 articles being included in the review.

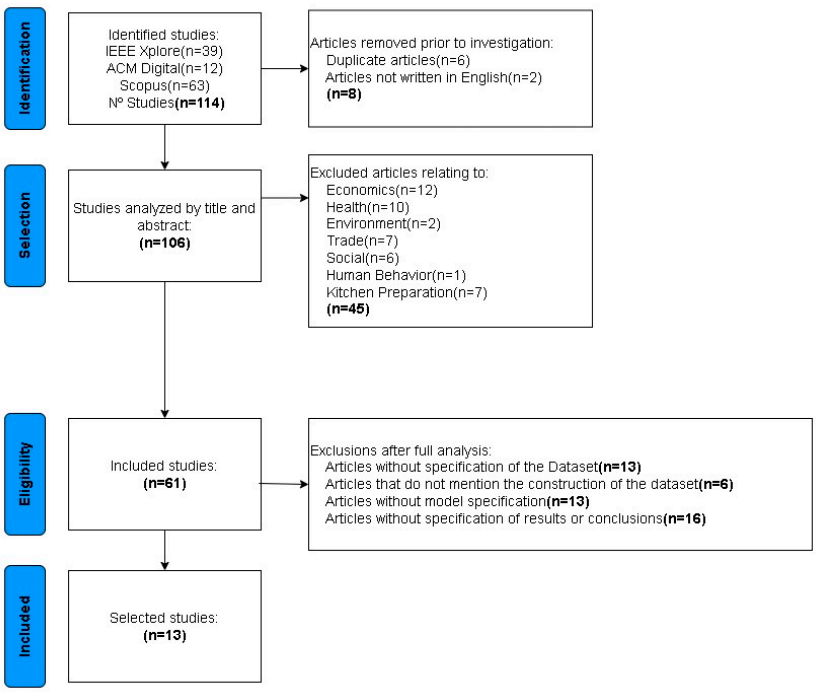

Figure 2. Flowchart of research phases.

Data were extracted from all the studies identified in the following format: study; model; description. Table 1 identifies the most important characteristics of the selected studies.

Table 1. Scientific articles analyzed.

Study	Model	Description
[14]	ResNet-50	The authors developed an algorithm to recommend recipes by recognizing ingredients in advance. They trained the ResNet-50 model over 20 epochs using transfer learning, and once the model was trained, it was used to recommend recipes. The model is capable of recognizing 32 ingredients, and thus, a 19 × 32 matrix was created in which the index is set to "1" when an ingredient is identified. Subsequently, a linear search is conducted on the database, resulting in the retrieval of a recipe with indexes set to "1" for the ingredients identified by ResNet-50 and "0" for the remaining ingredients.
[15]	EfficientNet-B2	The authors propose a food recognition system. They trained the model on ImageNet to extract generic features and used fine-tuning to obtain better results. The model weights were adjusted in the last two blocks and three additional blocks were added, each containing a fully connected layer and a dropout layer. Finally, data augmentation techniques were applied to further reduce overfitting. The authors demonstrated that the use of fine-tuning significantly improves the model's performance.
[16]	ResNet-50	The authors have developed a method that adapts to users' eating habits. Each user has their own database with their food records. Subsequently, a time-dependent food distribution model is employed, which takes into account the evolution of the user's eating habits over time. This is achieved through the use of a vector weight optimization strategy, which optimizes the weight of the classifier vectors and thus better adapts to changes in eating habits.
[17]	ResNet VGG19 EfficientNet-B0 DenseNet	The authors propose an approach to recognize food dishes and consequently recommend recipes. Each dish is represented in a matrix with index '0'. After the model assigns a class to the image, the information is extracted from web pages using the web crawling method with the Beautiful Soup and Selenium libraries. The models are trained for 50 epochs using ImageNet transfer learning, and data augmentation is also applied.
[18]	ResNet-50 Vision Transformer 16/B	The authors have developed a framework for classifying ingredients in images of food dishes. The first module, ReLeM (ReciPe Language-Enhanced Multimodal), is designed to enhance the accuracy of ingredient segmentation. To achieve this, the visual representations of ingredients that appear in various dishes are integrated with the recipe language, recognizing that the same ingredient can be represented in different ways due to different preparation methods. The second module is tasked with classifying the segmented zones. The images are processed through a vision encoder, which is then followed by a vision decoder. Finally, segmentation models are employed to identify the ingredients in the image that have previously been classified.
[19]	BEiT	The authors propose a food image recognition method based on generative self-supervised learning. The objective is to enable the model to be trained on unlabeled datasets, allowing it to make segmentation predictions and circumvent the high costs associated with hiring specialized teams. To achieve this, the authors utilize the BEiT model, which has been pre-trained on ImageNet, to reconstruct portions of the image that are not visible. Subsequently, fine-tuning is employed to adapt the network to the dataset utilized by the authors (Food-101).
[20]	InceptionResnet V2 Resnet50 Densenet169 Wiser	The authors developed a method that focused on the importance of the quality of the training data. They used the U2-Net algorithm, which was designed to remove background objects from food images and improve model performance. After training several models, it was concluded that removing the background and using data augmentation techniques helped to improve the accuracy rate.

Table 1. *Cont.*

Study	Model	Description
[21]	MultiTask Deep Belief Network (M3TDBN)	The authors propose a MultiTask Deep Belief Network (M3TDBN) that can identify ingredients in recipes through textual representation. The model considers various attributes related to the recipe, such as the type of cuisine and the type of dish. The Yummly-28K dataset, comprising 63,492 recipe images, was used to train the model. To identify the ingredients, it is necessary to pre-process the text, removing irrelevant information such as quantity. Transfer learning was employed using a convolutional neural network (CNN) pre-trained on the Food-101 dataset, which was then fine-tuned to the Yummly-28K dataset. The features extracted by this CNN were used as inputs for the M3TDBN model. It was concluded that incorporating additional information, such as the type of dish, significantly improved performance.
[22]	ResNet-50 ResNet-101 VGG-19	To address the challenge of ingredient recognition in food images, the authors propose a D-Mixup (Dynamic Mixup) approach. The objective is to enhance the representation of minority ingredients, given the prevalence of significant disparities in the frequency of certain ingredients across most datasets. Additionally, this method mitigates the issue of datasets where test images exhibit high similarity to training images, which may not accurately reflect their true effectiveness in real-world scenarios. This structure also comprises a region-wise recognition network, which is responsible for identifying the ingredients present in each region. The results of this approach indicate that it improves performance on datasets that present these problems.
[23]	ResNet-10 EfficientNet-B0	The authors propose a unified structure that encompasses two distinct approaches: many-shot learning and few-shot learning. Additionally, the structure employs a convolutional graph network (GCN) to capture relationships between different categories of food. To circumvent the issue of suboptimal performance in classes with limited image data (few-shot learning), this structure comprises two distinct phases. In the initial phase, semantic embeddings are generated utilizing the BERT (Bidirectional Encoder Representations from Transformers) model, thereby furnishing supplementary data on the specific type of food associated with each food category. Subsequent to this, a convolutional neural network (CNN) is employed to extract features from images comprising a multitude of samples, after which the integration of learning from numerous and few categories is achieved. In the second phase, the graph convolutional network (GCN) is integrated into the model to facilitate the comprehension of the interconnections and distinctions between categories. This approach has been demonstrated to yield a notable enhancement in performance compared to more sophisticated studies on few-category learning.
[24]	ResNet-50	The authors propose a method that employs a convolutional neural network (CNN) ResNet-50 for food recognition, utilizing the fine-tuning technique following the initial training of the model on the ImageNet database. The images are then subjected to pre-processing, including resizing to a resolution of 224x224 pixels, and convolutional filters are applied to extract the most relevant features. It is observed that the greater the number of classes, the lower the accuracy. Conversely, the greater the number of instances per class, the greater the accuracy.
[25]	VGG-16 Resnet-50 Mobilenet-V3 YOLOv5	The authors' approach to food recognition divides the problem into two topics: binary classification and locating the food followed by its category. Binary classification identifies whether an image contains food or not, while food localization usually requires the use of bounding boxes, which are then used to classify several foods in the same image using convolutional neural networks (CNNs).

Table 1. *Cont.*

Study	Model	Description
[26]	ResNet-101 GoogleNet VGG16/19 InceptionV3	The authors propose an approach to food recognition that combines the results of different convolutional neural networks (CNNs) to determine the class of food using ensemble methods based on voting. There are two approaches to this type of method: hard voting and soft voting. In the first approach, each model makes a prediction, and the class with the most votes is assigned. In the second approach, the probability of each class is used and averaged, with the class with the highest value being chosen. Additionally, a Bayesian optimization algorithm is employed to identify the optimal weight for each CNN, considering its accuracy.

A review of the selected articles reveals a multitude of methods for recognizing foods or ingredients. All of these articles attempt to mitigate this problem, with some implementing more complex approaches that allow for the segmentation and identification of ingredients, resulting in improved accuracy in calorie estimation tasks, which are essential in food control applications.

Regarding the period of the studies analyzed, it can be observed that they were published between 2016 and 2023, with the majority published between 2021 and 2023, as illustrated in Figure 3.

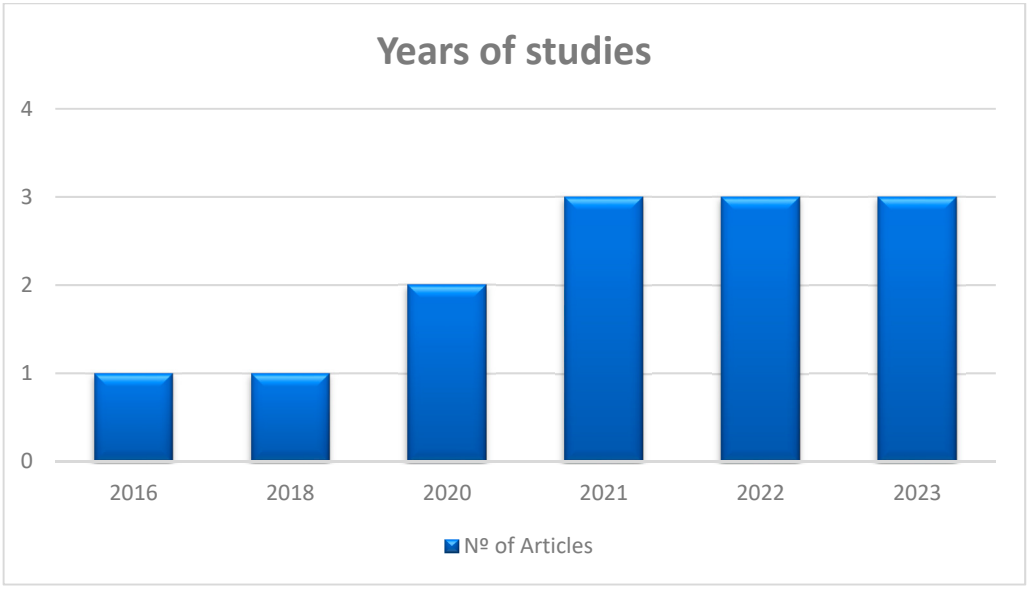

Figure 3. Date of publication of the articles.

The authors of the articles analyzed predominantly employed convolutional neural network (CNN) architectures, as they represent the most advanced method for recognizing food in images. This type of network comprises a sequence of filters applied to the input image to extract its characteristics [27]. As illustrated in Figure 4, only 14% of the articles implemented a different architecture, indicating that only two studies utilized a distinct artificial neural network (ANN), namely deep neural network (DNN) and graph neural network (GNN). The remaining studies employed CNNs with distinct models for food recognition.

Figure 5 depicts the models utilized in the studies, demonstrating that there are numerous models capable of classifying food images, with varying degrees of effectiveness. This elucidates the high diversity observed, as numerous authors perform a limited benchmark with select models to ascertain which one performs optimally. The most widely used model was ResNet-50, which will be employed to run the tests on the selected dataset. This model has been selected due to its demonstrated ability to achieve SOTA accuracy on the Ima-

geNet dataset, as evidenced by the literature available on the web [28]. Given that transfer learning will be employed on this dataset, this model is considered the optimal choice.

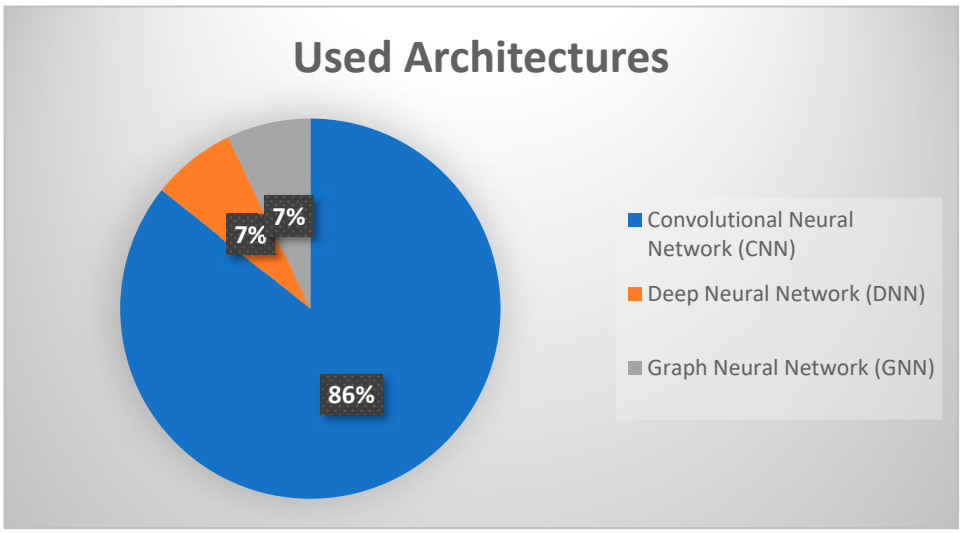

Figure 4. Percentages of the architectures used in the articles.

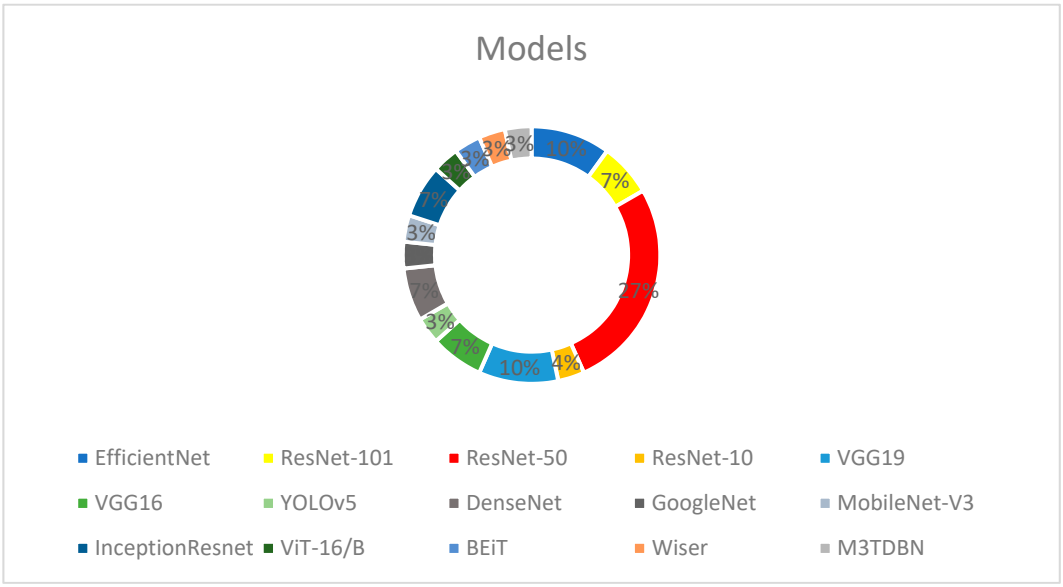

Figure 5. Percentages of the models used in the articles.

As with the models employed, the authors also frequently utilize a variety of datasets to substantiate the efficacy of the architectures and draw inferences regarding the optimal dataset for their implementation. As illustrated in Figure 6, there is a notable prevalence of the utilization of the Food-101 dataset for food recognition.

After investigating the selected studies, it can be concluded that it is essential to analyze several components, including the network to be used, the model, and the dataset. Within each of these components, several aspects must be considered to ensure the success of network training. Unfortunately, most articles only recognize food, with recipe recommendations being a much rarer approach. However, we have managed to select some articles with very interesting approaches to achieving this.

Based on the information obtained, the following answer to the research questions is presented:

(a) Question 1: Are there digital solutions that recognize food from images of meals?

Food recognition is a proven advantage for making the eating process simpler, more economical, and healthier. Various solutions can be applied to this task using artificial intelligence and machine learning algorithms, as mentioned in the articles [14–26]. All of the aforementioned articles are capable of recognizing food through a variety of approaches, resulting in varying degrees of accuracy.

(b) Question 2: Are there digital solutions that propose recipes based on the re-recognition of ingredients or food images?

Recipe recommendation is a valuable approach for many recipe management applications that do not yet utilize real-time food recognition mechanisms. The integration of this type of technology will streamline the process of meal decision-making. Although many of the models implemented for food recognition can be used for recipe recommendation, the majority of articles do not explicitly refer to this process. However, in articles [14,17,21], different approaches are used for this purpose.

(c) Question 3: Are there solutions for proposing meals based on leftovers, taking into account the characteristics of each user?

The issue of food waste has gained prominence in recent years, driven by the significant increase in consumerism and food diversity. This phenomenon, which has been growing at an alarming rate, has become a major issue, with consequences in the social, economic, and environmental spheres. Study [16] mentions a method that adapts to the user's eating habits. However, it is important to note that more responsible and healthier practices can be adopted through approaches that recommend recipes.

Based on the analysis of the proposed studies, it was possible to conclude that the most widely used model was ResNet-50 and the most used dataset was Food-101. The next section will assess whether this combination can be used since the Food-101 dataset is very complex and the model may not perform satisfactorily.

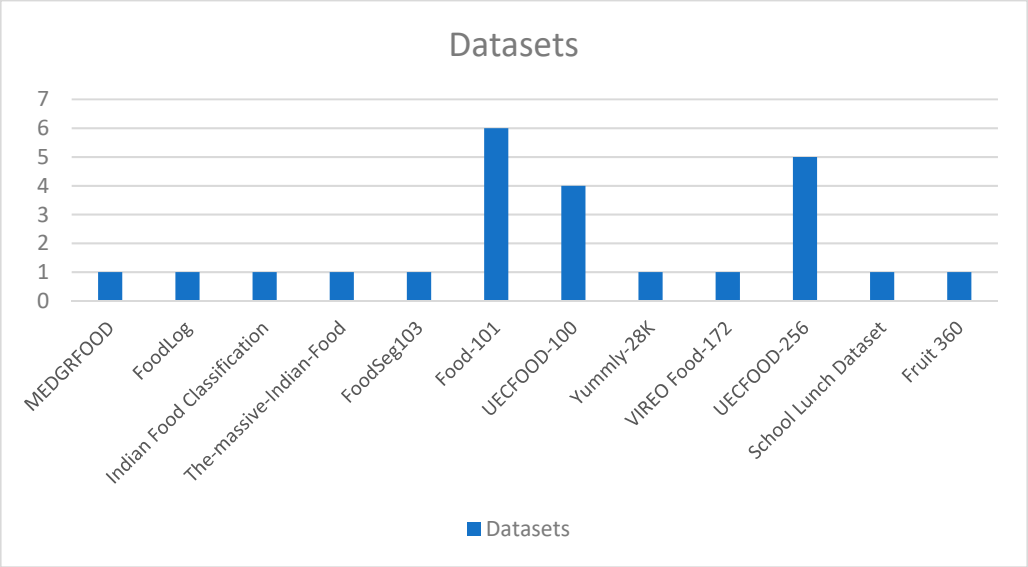

Figure 6. Quantity of datasets used in articles.

3. Materials and Methods

The aim of this chapter is to understand the architecture of the ResNet-50 model and why it should not be applied to the Food-101 dataset, since it has serious limitations that will be described. Consequently, the process of implementing a novel dataset through web scraping using the Selenium library and ChromeDriver will be delineated, thus overcoming the limitations of Food-101. All tests were conducted on a personal computer (HP ENVY ×360 15.6-inch) with a Core i7-1165G7 central processing unit (CPU), 16 GB of DDRAM, and a 512 GB solid-state drive, running the Windows 11 Home operating system.

3.1. ResNet-50 and Food-101 Features

The ResNet-50 model comprises 50 layers, including 48 convolutional layers, 1 MaxPool layer, and 1 Average Pool layer. The architecture is divided into four parts: the convolutional layers, the identity block, the convolutional block, and the fully connected layers. The convolutional layers extract the characteristics of the image. Batch normalization and ReLU activation are then applied. Subsequently, max-pooling layers are employed to reduce the size of the extracted feature map while preserving the most crucial features [29]. The identity block comprises three layers, wherein the initial and concluding layers possess an identical number of filters and are utilized when the input dimensions align with the output dimensions, while the convolutional block is employed when the dimensions differ [30]. However, it is in these two blocks that the skip connection concept is applied, allowing the model to skip some layers of the network and feed the output of one layer as an input to the following layers [31]. Finally, the last part of the ResNet-50 model consists of fully connected layers that make the final prediction. The number of neurons in this layer is equal to the number of existing classes.

The Food-101 dataset comprises 101,000 food images, distributed across 101 classes. Each class in the training group contains 1000 images, while the validation group contains 150 images.

3.2. Proposed Model for Ingredient Identification

The ResNet-50 model was trained on the Food-101 dataset using the standard architecture, with only the output layer configured to have the same number of neurons as the classes in Food-101. However, this approach presents a significant challenge, as it is not possible to train the model without first modifying the dimensions of the images. The default size of the images in the dataset (512 $\times$ 512 pixels) resulted in an error on the Kaggle platform due to excessive memory allocation. Therefore, it was essential to resize the images to 224 $\times$ 224 pixels. This was only necessary due to the complexity of the Food-101 dataset and the limitations imposed by the Kaggle platform itself. The model was trained for 50 epochs with a batch size of 128 without the use of transfer learning.

The compilation of this model produced suboptimal results, as depicted in Figure 7, which unambiguously represents the phenomenon of overfitting. The model has learned the characteristics of the training images but cannot generalize to unseen images.

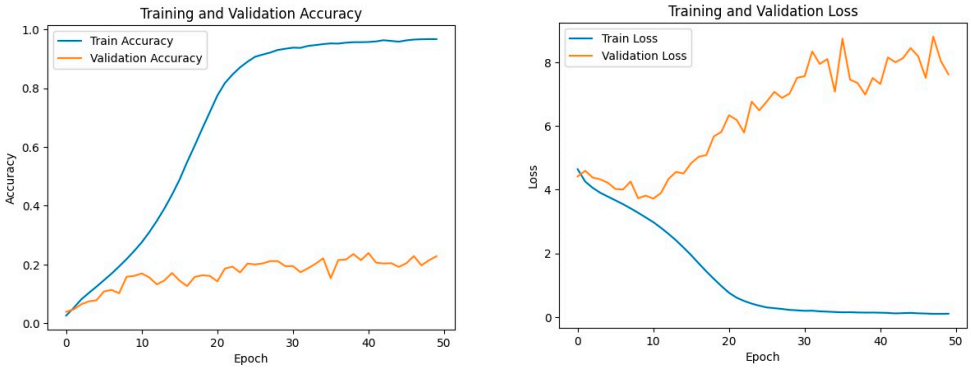

Figure 7. ResNet-50 results on Food-101 dataset.

Several modifications were made to the model to ascertain whether it would be possible to achieve the requisite degree of accuracy to justify its use in a future mobile application. The proposed architecture commences with the construction of a sequential model, whereby the layers are added one after the other in sequence. Subsequently, the ResNet-50 model, which has been pre-trained on the ImageNet database, is incorporated into the model. The following layers are then added in sequence: The flatten layer transforms the 3D feature vector produced by the pre-trained ResNet-50 model into a 1D vector, as the subsequent layers require inputs of this type. The dense layer employs the ReLU activation function and both L1 and L2 (elastic net) regularizations, whereby L1 selects the most relevant

features and L2 prevents overfitting [28]. The dropout layer randomly disconnects the neurons from the previous layer, and the final dense layer has several neurons equal to the number of classes in the dataset. However, it maintains the L1 and L2 regularization, which is appropriate for multi-class classification problems. Finally, the model is trained over 40 epochs with a learning rate of 0.0001.

Subsequently, the final 10 layers of the ResNet-50 model are designated as trainable (fine-tuning), and the LearningRateScheduler callback is configured. This entails that at each epoch, the learning rate is updated by the schedule function defined in [32].

Figure 8 illustrates the various stages of the ResNet-50 model. It is crucial to emphasize the data augmentation stage, during which a range of techniques, including rotation and brightness alteration, are employed to ensure that the model can generalize to new images. Subsequently, the pre-trained ImageNet weights (transfer learning) are utilized, enabling the model to be trained on a vast dataset and to learn the characteristics of images in general. Finally, the model is fine-tuned on the Food-101 dataset, allowing it to adapt to the specific characteristics of the images in the context of the problem.

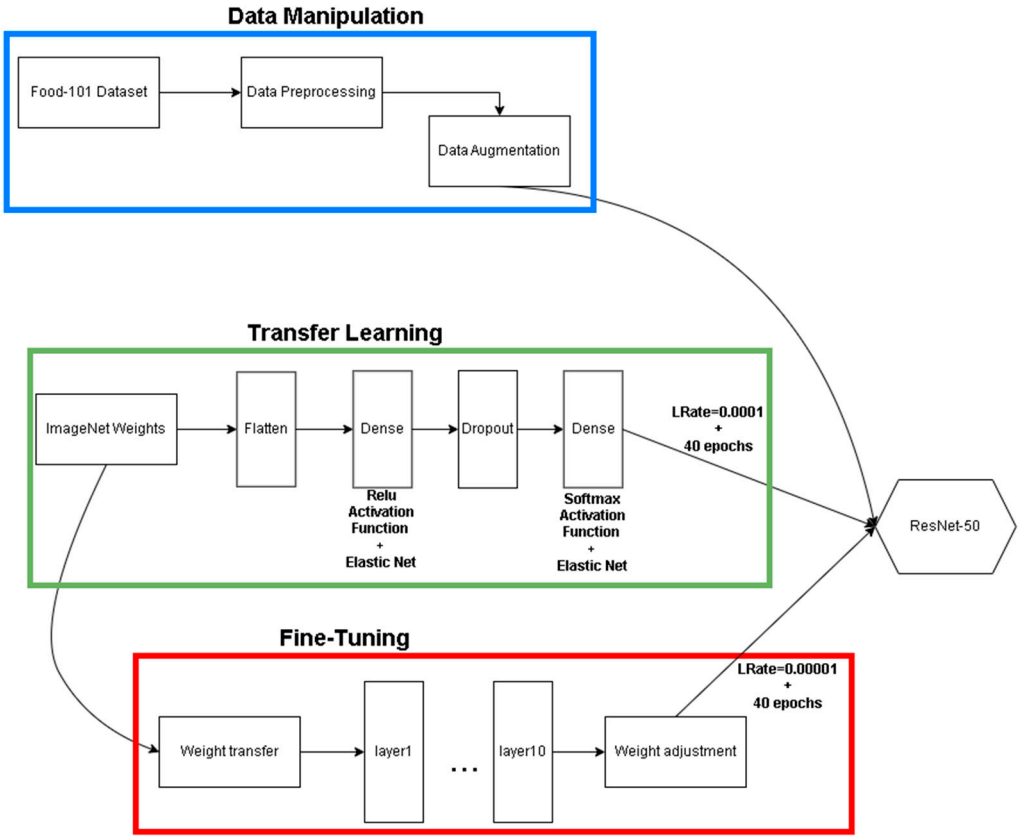

Figure 8. ResNet-50 model using transfer learning and fine-tuning.

To assess the impact of cropping the images on the model's performance, the model was trained on a small subset of the Food-101 dataset. This subset included 20 classes with 100 images in the training group and 20 in the validation set.

Figure 9 illustrates the model's accuracy when the images are resized to 512 pixels, while in Figure 10, the accuracy is associated with 224 × 224 pixel images. Although the model's accuracy is significantly higher, it is still not sufficient to guarantee the model's reliability. This is because the model's accuracy for the validation set, a collection of images not seen during training, is approximately 70%. This means that if 100 accuracies are made, the model will incorrectly classify 30 foods. Consequently, using this model would have serious consequences for the mobile application that will be implemented. In addition to the ability to recognize foods, the application must also be able to recommend person-

alized recipes. Therefore, an incorrect classification of foods would result in inadequate recommendations, which could hurt users' health and their trust in the system.

Figure 9. Precision with 512 pixel images.

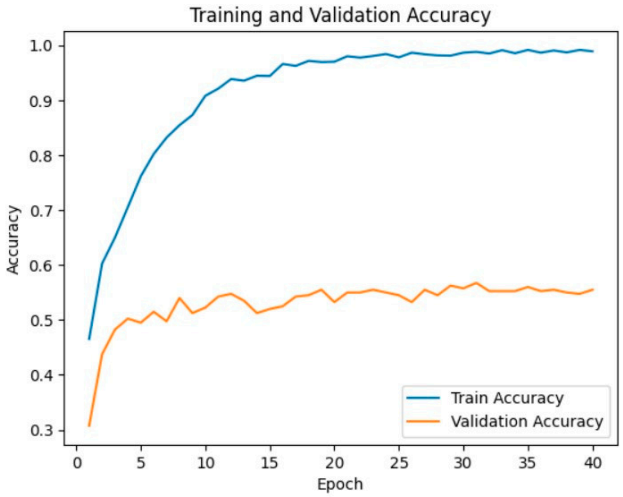

Figure 10. Precision with 224 pixel images.

The figures illustrate the accuracy of the model trained on the entire Food-101 dataset with images cropped to a size of 224 × 224 pixels. This accuracy justifies the initial results obtained by the model, despite the difference in validation accuracy of 15% between the two datasets. This difference would be substantially greater if it were possible to train the entire dataset with the original resolution.

3.3. Food-101 Problems and Alternatives

The suboptimal performance of the model is primarily attributable to the substantial storage requirements of the images in the Food-101 dataset (512 × 512 pixels), which necessitate cropping and result in the loss of information, as illustrated in Figure 11. Given that this dataset occupies 10 GB of memory and is quite complex to be trained in several epochs, it is more suitable to be used as transfer learning, as was the case in one of the articles [21].

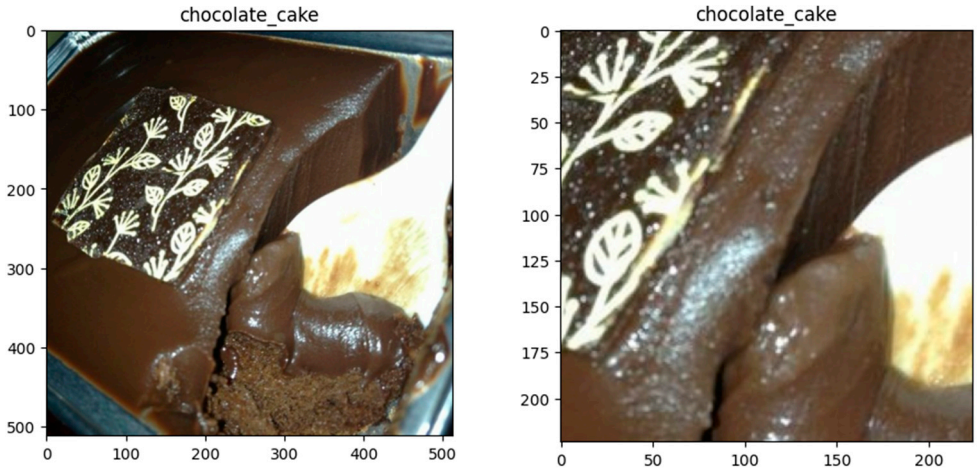

Figure 11. Cropping images from the food-101 dataset.

3.4. Comparison and Evaluation of Results

As previously stated in [15], the Food-101 dataset should be employed to assess the efficacy of deep learning models, as it was designed to represent a realistic food classification challenge, encompassing images captured under a range of lighting conditions, angles, and contexts. This enables the testing of the robustness of image classification models.

It is not uncommon for authors to utilize a subset of this dataset for classification models to classify food. However, it should be noted that the images in this dataset were not created for this purpose. Rather, they were designed to evaluate the capacity of machine learning models. For instance, in article [14], the authors utilize only the ice cream and onion classes and subsequently integrate them into other datasets. Conversely, in article [20], they employ only 16,256 images belonging to 17 classes. In articles [23,24], the images are also resized, but the accuracy achieved by the authors is significantly lower, reaching only around 56% and 41% accuracy.

3.5. New Dataset

The Selenium Python library [33] was used to build the new dataset. This library enables programmers to interact with web pages by simulating user actions. Consequently, it is frequently employed in conjunction with ChromeDriver, which permits the library to interact with the Google Chrome browser.

The objective of our dataset was not merely to include images of foods from various parts of the meal (e.g., main course, fruit, dessert) but also to enhance the global accessibility of these foods. Open-source datasets often represent cuisines such as Thai, Turkish, or Vietnamese, which are not widely consumed outside of their respective regions.

A dataset search was conducted, and this was constructed by joining four datasets available on Kaggle: the "Food-11 image dataset", the "MAFood-121", the "food-101-tiny", and the "Fruits Classification". The combination of these datasets resulted in 30 classes with a scarce and unbalanced number of images for each class since they were taken from different sources. To circumvent this issue, a Python script was developed to download images from the Google browser. The download was conducted for the Google, Bing, and Yahoo search engines, from which 500, 250, and 50 images were obtained, respectively. The search engines were selected because they are currently the most widely used, according to [34]. As Bing and Yahoo are not optimal for downloading a large number of images, it was determined that a smaller number of images would have to be downloaded.

Figure 12 illustrates part of the code necessary to download the images. This script comprises three principal functions: main, fetch_image_urls, and persist_image. The main function establishes a connection to the driver and inputs the search queries into the search box. fetch_image_url then extracts the links to the images, which are subsequently stored locally by the persist_image function. To execute this script on various search engines, it is

necessary to modify the CSS selectors (used to identify elements on a web page) as each site employs a distinct HTML structure and CSS elements.

```
if __name__ == '__main__':

    service = Service(executable_path=DRIVER_PATH)
    options = webdriver.ChromeOptions()
    wd = webdriver.Chrome(service=service, options=options)

    queries = ["Apple"]

    n=0
    for query in queries:
        wd.get('https://google.com')
        if n==0:
            cookie_button = WebDriverWait(wd, 10).until(
            EC.presence_of_element_located((By.ID, 'L2AGLb')))
            cookie_button.click()
            n+=1

        wait = WebDriverWait(wd, 10)
        search_box = wait.until(EC.element_to_be_clickable((By.CSS_SELECTOR, 'textarea.gLFyf')))
        search_box.send_keys(query)
        links = fetch_image_urls(query, 50, wd)
        images_path = '/Users/joaod/Desktop/imagens'
        for i in links:
            persist_image(images_path, query, i)
    wd.quit()
```

Figure 12. Web scraping with ChromeDriver.

As previously stated, this process may result in the retrieval of images that are not suitable for the intended purpose. To ensure the reliability of the dataset, each of the folders was manually filtered. Following this process, the resulting dataset comprises 30 classes, 26,709 images for training, and 5311 for validation. Each of the classes used for training comprises approximately 900 images, while those used for validation comprise around 200. Figure 13 illustrates the classes of which the dataset is made, demonstrating a considerable diversity of foods, the majority of which are known to the European public.

Figure 13. Folders of the dataset classes.

We can characterize our dataset as follows:

- 30 food classes;
- 32,020 images;
- Food, starter, or dessert images;
- Image format jpg;
- Images of different ways of cooking food.

3.6. Results and Discussion

The objective at this stage is to ascertain the viability of utilizing ResNet-50 in our dataset, with a view to its deployment in the real world.

The process that led to the final configuration of the dataset, and in particular the ResNet-50 model, necessitated the execution of a series of tests and the assessment of various hyperparameters. The optimal outcome was achieved with a model trained for 40 epochs. These metrics are commonly employed for the evaluation of machine learning problems. Accuracy represents the proportion of correct predictions, although it can be misleading in the context of imbalanced datasets, where there are many more examples of one class than another; our dataset does not have this problem. In contrast, a function that evaluates the model's predictions and the true target values is employed to calculate the loss, which depends on the type of problem. In our case, we used CategoricalCrossentropy. Figure 14 illustrates the training and validation accuracy, which were approximately 97% and 90%, respectively. Figure 15 depicts the training and validation errors, which were approximately 2% and 6%, respectively.

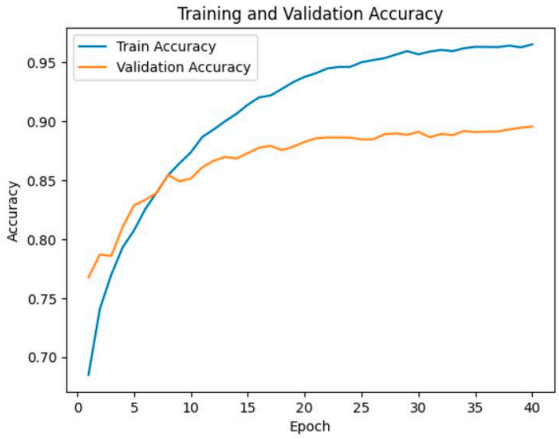

Figure 14. Precision graph.

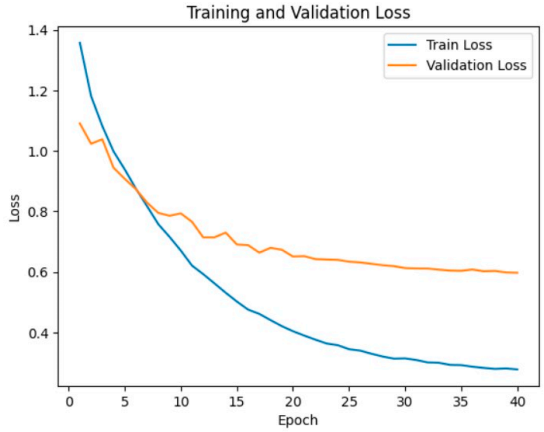

Figure 15. Loss graph.

To evaluate the model's performance, a confusion matrix was constructed, which enabled the identification of inclusion and exclusion errors and the determination of which classes the model had the greatest difficulty identifying [35].

Figure 16 illustrates the confusion matrix for the ResNet-50 model implemented on the dataset created for this study. It can be observed that the numbers on the main diagonal represent the vast majority of the validation set, indicating that the model correctly classified the majority of samples. Despite exhibiting minimal error, the two classes for which the model exhibited the least accuracy were ramen and egg. In some instances, the model confused these classes with soup and omelet, respectively. This is understandable given that these classes share numerous similarities in visual appearance.

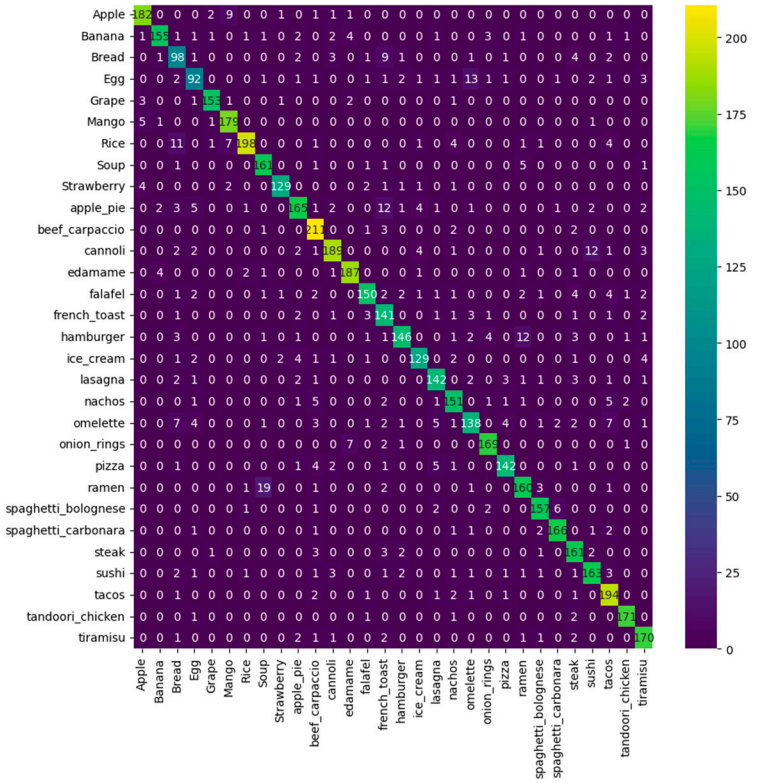

Figure 16. Confusion matrix.

As previously stated, the accuracy metric is not always a reliable indicator of model performance. Consider a fictitious dataset with two classes, with 90 samples for the first class and 10 samples for the second. If the model classifies all instances as the first class, the accuracy will be 90%.

Following the calculation of the confusion matrix, it is essential to assess the F1 score, which is based on the precision and recall values.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F1 Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

This metric employs the harmonic mean instead of the simple arithmetic mean [36], thus evaluating both false positives (low precision) and false negatives (low recall), rendering it a robust metric. Figure 17 illustrates the F1 score for each class, with all values exhibiting a high degree of accuracy, indicating that the model's predictions are not particularly precise.

To evaluate the model's effectiveness using real-world photos, a final test was conducted using food from a home environment. The purpose of this test is to graphically depict the performance of the model by including the likelihood of the top 5 foods that the model assigned the highest score, regarding the food depicted in the image. The findings indicate that ResNet-50 demonstrates exceptional performance on the custom dataset de-

veloped by the authors and exhibits the capacity to generalize to images that are not part of the training set. This is demonstrated in Figures 18 and 19, where ResNet-50 accurately assigns a probability of 1.0 to all food items.

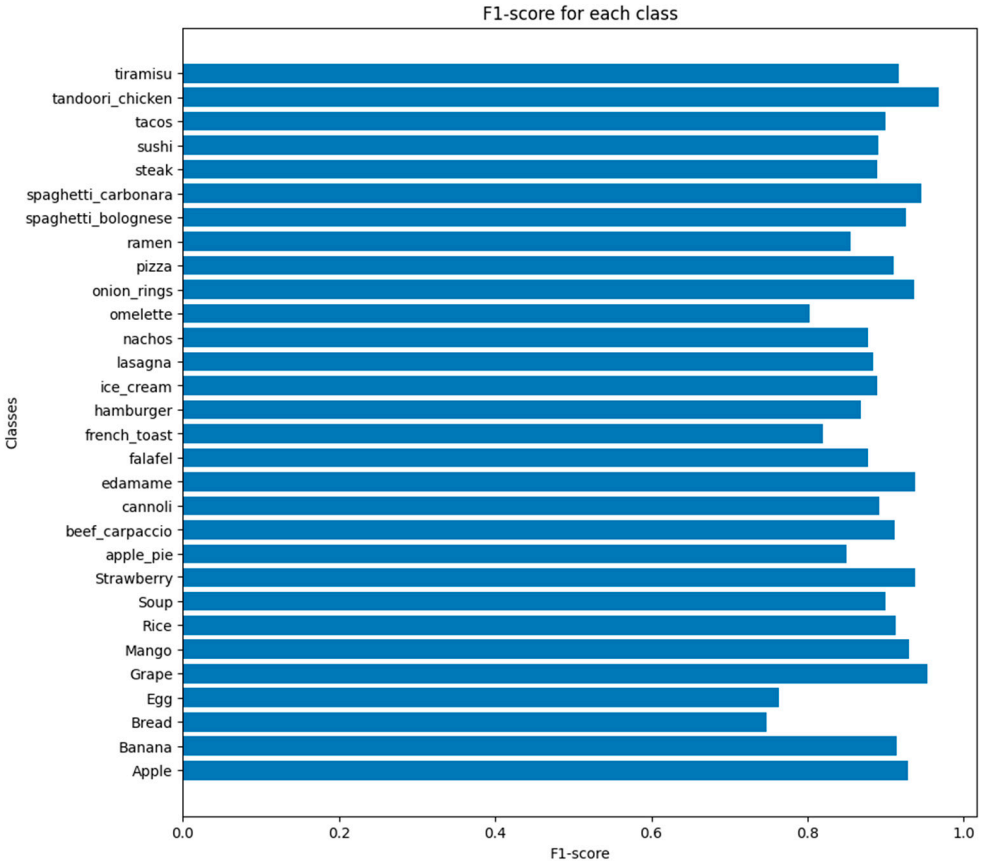

Figure 17. F1 score.

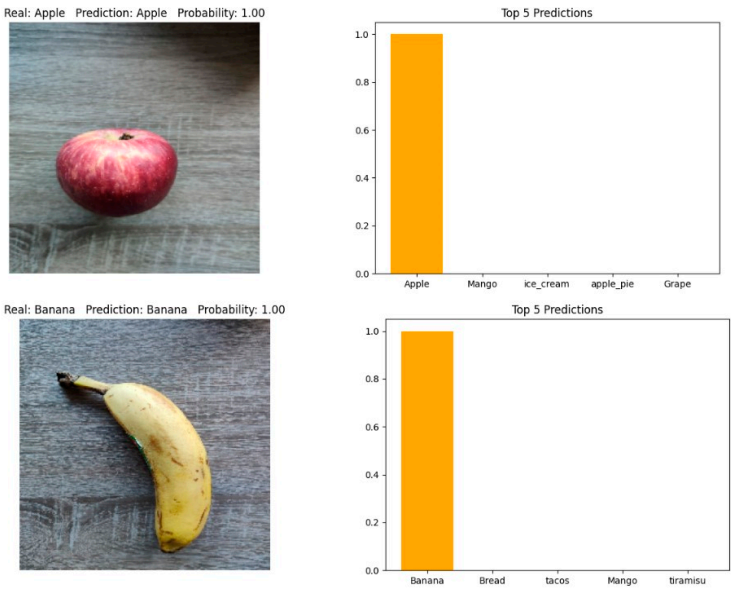

Figure 18. 1º Test with food from home.

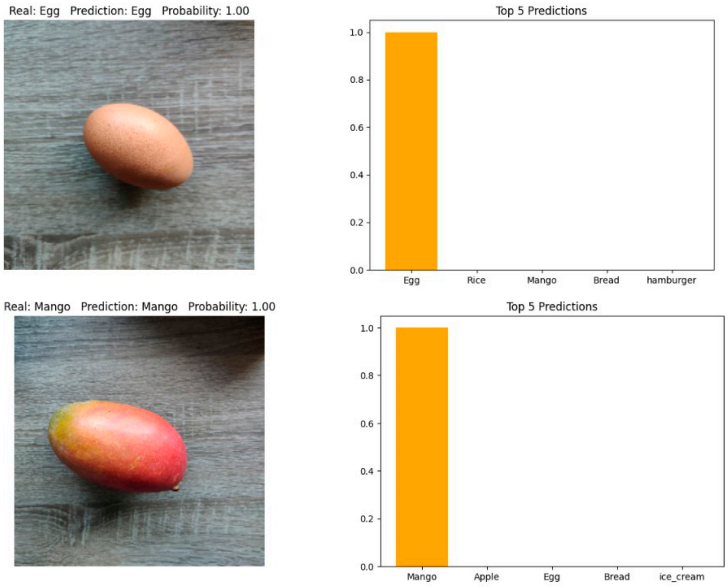

Figure 19. 2° Test with food from home.

4. Conclusions

This article presents an in-depth analysis of various studies in the field of food recognition technology, employing state-of-the-art methodologies and approaches. After an analysis of the configuration used by other authors, it is concluded that the ResNet-50 model is currently the most widely used, in conjunction with the Food-101 dataset. Consequently, the characteristics of this model and the Food-101 dataset have been analyzed and explained in detail.

Initially, ResNet-50 was utilized without any modifications. However, due to overfitting, a convolutional neural network (CNN) ResNet-50 was constructed layer by layer. These alterations included the utilization of transfer learning using ImageNet, as well as the modification of hyperparameters such as the number of epochs, the batch size, and the learning rate, to maximize the model's accuracy. This resulted in a significant reduction in overfitting and an increase in validation accuracy. Subsequently, several tests were conducted to ascertain the reasons behind the model's suboptimal accuracy. The investigation revealed that the issue originated from the dataset itself. To identify the underlying cause, we examined the impact of the cropping process on the images, which was constrained by the hardware limitations of Kaggle. Our findings indicated that this cropping procedure resulted in a significant loss of information, which ultimately compromised the model's performance. Given the complexity of the images and the size of the Food-101 dataset, which is a large collection of images, we recommend the use of transfer learning, as is implemented in the ImageNet dataset.

To circumvent the issues associated with the Food-101 dataset, we employed the web scraping technique utilizing the Selenium library and ChromeDriver, resulting in a dataset comprising 32,020 images, which is balanced and diverse. Subsequently, we conducted a series of tests on the model, including accuracy, confusion matrix, F1 score, and tests with homemade food, to comprehensively evaluate the model's capabilities. Training the model on our dataset obviated the necessity to crop the images, thus circumventing the issue of differing accuracies observed in the Food-101 dataset between the original images and those resized to 224 pixels. All images in our dataset are smaller than 300×300 pixels.

For future work, it would be advantageous to extend the dataset to include more foods, thus avoiding the unnecessary waste of food. Furthermore, it would be beneficial to implement this configuration, namely the ResNet-50 model applied to the dataset created, in a mobile application so that the user can receive feedback on potential recipes that can be made, using a database or an application programming interface (API).

In essence, the journey to optimize food recognition systems necessitates the integration of sophisticated methodologies, meticulous selection of datasets, and strategic implementation of convolutional neural network (CNN) techniques. The adoption of this knowledge will facilitate the advancement of transformative techniques for the reduction of food waste and the promotion of sustainable culinary practices.

Author Contributions: Conceptualization, methodology, J.L., F.F. and Â.O.; software, validation, investigation, resources, data curation, writing—original draft, J.L.; writing—review and editing, J.L., F.F. and Â.O.; supervision, F.F. and Â.O. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: This research received no external funding.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Acknowledgments: This work was funded by National Funds through the Foundation for Science and Technology (FCT), I.P., within the scope of the project UIDB/05583/2020 and DOI identifier https://doi.org/10.54499/UIDB/05583/2020. Furthermore, we would like to thank the Research Centre in Digital Services (CISeD) and the Instituto Politécnico de Viseu for their support.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Recommendation Systems: Applications and Examples in 2024. Available online: https://research.aimultiple.com/recommendation-system/ (accessed on 30 May 2024).
2. Best Recipe Apps: The 7 Finest Apps for Cooking Inspiration | TechRadar. Available online: https://www.techradar.com/news/best-recipe-apps-the-7-finest-apps-for-cooking-inspiration (accessed on 30 May 2024).
3. Spoonacular Recipe and Food API. Available online: https://spoonacular.com/food-api (accessed on 5 June 2024).
4. Edamam—Food Database API, Nutrition API and Recipe API. Available online: https://www.edamam.com/ (accessed on 5 June 2024).
5. TensorFlow. Available online: https://www.tensorflow.org/?hl=pt-br (accessed on 17 January 2024).
6. Keras: Deep Learning for Humans. Available online: https://keras.io/ (accessed on 17 January 2024).
7. NumPy. Available online: https://numpy.org/ (accessed on 17 January 2024).
8. What Is a Dataset? Definition, Use Cases, Benefits, and Example | by Bright Data | Medium. Available online: https://medium.com/@Bright-Data/what-is-a-dataset-definition-use-cases-benefits-and-example-9aaf5ecc301e (accessed on 5 June 2024).
9. Why Web Scraping: A Full List of Advantages and Disadvantages | by Teodora C. | Medium. Available online: https://raluca-p.medium.com/why-web-scraping-a-full-list-of-advantages-and-disadvantages-fdbb9e8ed010 (accessed on 5 June 2024).
10. PRISMA. Available online: http://www.prisma-statement.org/?AspxAutoDetectCookieSupport=1 (accessed on 17 January 2024).
11. IEEE Xplore. Available online: https://ieeexplore.ieee.org/Xplore/home.jsp (accessed on 17 January 2024).
12. Scopus—Document Search. Available online: https://www.scopus.com/search/form.uri?display=basic#basic (accessed on 17 January 2024).
13. ACM Digital Library. Available online: https://dl.acm.org/ (accessed on 17 January 2024).
14. Morol, M.K.; Rokon, M.S.J.; Hasan, I.B.; Saif, A.M.; Khan, R.H.; Das, S.S. Food Recipe Recommendation Based on Ingredients Detection Using Deep Learning. In *Proceedings of the 2nd International Conference on Computing Advancements, Dhaka, Bangladesh, 10–12 March 2022*; ACM International Conference Proceeding Series; Association for Computing Machinery: New York, NY, USA, 2022; pp. 191–198. [CrossRef]
15. Konstantakopoulos, F.S.; Georga, E.I.; Fotiadis, D.I. Mediterranean Food Image Recognition Using Deep Convolutional Networks. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, Virtual, 1–5 November 2021; pp. 1740–1743. [CrossRef]
16. Yu, Q.; Anzawa, M.; Amano, S.; Ogawa, M.; Aizawa, K. Food Image Recognition by Personalized Classifier. In Proceedings of the International Conference on Image Processing, ICIP, Athens, Greece, 7–10 October 2018; pp. 171–175. [CrossRef]
17. Basrur, A.; Mehta, D.; Joshi, A.R. Food Recognition using Transfer Learning. In Proceedings of the IBSSC 2022—IEEE Bombay Section Signature Conference, Mumbai, India, 8–10 December 2022. [CrossRef]
18. Wu, X.; Fu, X.; Liu, Y.; Lim, E.P.; Hoi, S.C.H.; Sun, Q. A Large-Scale Benchmark for Food Image Segmentation. In Proceedings of the MM 2021—Proceedings of the 29th ACM International Conference on Multimedia, Virtual, 20–24 October 2021; pp. 506–515. [CrossRef]

19. Zhu, S.; Ling, X.; Zhang, K.; Niu, J. Food Image Recognition Method Based on Generative Self-supervised Learning. In *Proceedings of the 2023 9th International Conference on Computing and Artificial Intelligence, Tianjin, China, 17–20 March 2023*; ACM International Conference Proceeding Series; Association for Computing Machinery: New York, NY, USA, 2022; pp. 203–207. [CrossRef]
20. Raman, T.; Kumar, S.; Paduri, A.R.; Mahto, G.; Jain, S.; Bindhu, K.; Darapaneni, N. CNN Based Study of Improvised Food Image Classification. In Proceedings of the 2023 IEEE 13th Annual Computing and Communication Workshop and Conference, CCWC 2023, Las Vegas, NV, USA, 8–11 March 2023; pp. 1051–1057. [CrossRef]
21. Min, W.; Jiang, S.; Sang, J.; Wang, H.; Liu, X.; Herranz, L. Being a supercook: Joint food attributes and multimodal content modeling for recipe retrieval and exploration. *IEEE Trans. Multimed.* **2017**, *19*, 1100–1113. [CrossRef]
22. Gao, J.; Chen, J.; Fu, H.; Jiang, Y.G. Dynamic Mixup for Multi-Label Long-Tailed Food Ingredient Recognition. *IEEE Trans. Multimed.* **2023**, *25*, 4764–4773. [CrossRef]
23. Zhao, H.; Yap, K.H.; Kot, A.C. Fusion learning using semantics and graph convolutional network for visual food recognition. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision, WACV 2021, Waikoloa, HI, USA, 3–8 January 2021; pp. 1710–1719. [CrossRef]
24. Zahisham, Z.; Lee, C.P.; Lim, K.M. Food Recognition with ResNet-50. In Proceedings of the IEEE International Conference on Artificial Intelligence in Engineering and Technology, IICAIET 2020, Kota Kinabalu, Malaysia, 26–27 September 2020. [CrossRef]
25. Tan, S.W.; Lee, C.P.; Lim, K.M.; Lim, J.Y. Food Detection and Recognition with Deep Learning: A Comparative Study. In Proceedings of the International Conference on ICT Convergence, Melaka, Malaysia, 23–24 August 2023; pp. 283–288. [CrossRef]
26. Tasci, E. Voting combinations-based ensemble of fine-tuned convolutional neural networks for food image recognition. *Multimed. Tools Appl.* **2020**, *79*, 30397–30418. [CrossRef]
27. A Deep Convolutional Neural Network for Food Detection and Recognition | IEEE Conference Publication | IEEE Xplore. Available online: https://ieeexplore.ieee.org/document/8626720 (accessed on 18 January 2024).
28. How to Achieve SOTA Accuracy on ImageNet with ResNet50 | Deci. Available online: https://deci.ai/blog/resnet50-how-to-achieve-sota-accuracy-on-imagenet/ (accessed on 31 May 2024).
29. Deep Residual Networks (ResNet, ResNet50) 2024 Guide—Viso.ai. Available online: https://viso.ai/deep-learning/resnet-residual-neural-network/ (accessed on 18 January 2024).
30. Detailed Explanation of Resnet CNN Model. | by TANISH SHARMA | Medium. Available online: https://medium.com/@sharma.tanish096/detailed-explanation-of-residual-network-resnet50-cnn-model-106e0ab9fa9e (accessed on 20 January 2024).
31. What Are Skip Connections in Deep Learning? Available online: https://www.analyticsvidhya.com/blog/2021/08/all-you-need-to-know-about-skip-connections/ (accessed on 20 January 2024).
32. LearningRateScheduler | Tensorflow LearningRateScheduler. Available online: https://www.analyticsvidhya.com/blog/2021/06/decide-best-learning-rate-with-learningratescheduler-in-tensorflow/ (accessed on 2 June 2024).
33. Selenium with Python—Selenium Python Bindings 2 Documentation. Available online: https://selenium-python.readthedocs.io/ (accessed on 2 June 2024).
34. 11 Top Search Engines to Optimize For in 2024. Available online: https://www.oberlo.com/blog/top-search-engines-world (accessed on 2 June 2024).
35. What Is the Confusion Matrix? Available online: https://h2o.ai/wiki/confusion-matrix/ (accessed on 2 June 2024).
36. F1 Score in Machine Learning: Intro & Calculation. Available online: https://www.v7labs.com/blog/f1-score-guide (accessed on 2 June 2024).