*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 9***
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

# Fast and Accurate Calorie Count Prediction from Food Images using a Convolution Neural Network Method

**Mr. S. Mohideen Pillai**
Research Scholar
Department of Information Technology
Sri Ram Nallamani Yadava College of Arts and Science, Tenkasi
Affiliation of Manonmaniam Sundaranar University
Abishekapatti, Tirunelveli, Tamil Nadu, India
deenmca@gmail.com

**Dr. S. Kother Mohideen**
Associate Professor & Head
PG Department of Information Technology
Sri Ram Nallamani Yadava College of Arts and Science
Tenkasi, Tamil Nadu, India
drskmkother@gmail.com

**Abstract**—In the past ten years, with advances in deep learning techniques, automated object recognition has come very near to human levels of accuracy. Fast, automatic, and consistent image-based food calorie calculation is becoming a requirement with the increasing growth of overweight and other lifestyle- related disorders throughout the globe. Accurate and insightful solutions may be provided in the form of a mobile app with the aid of a deep learning-based automatic object recognition system. However, real-time image processing necessitates a high level of processing speed, which is a major consideration for such applications. Although several researchers have looked at estimating calories based on pictures of food, there is currently no image-driven, lightweight, rapid, and accurate food calorie estimation method. In this study, we offer a technique for recognizing common meals captured with a mobile phone camera by using Convolution Neural Networks (CNN) with optimum parameters. Calories and other nutritional information may be deduced from the known food class after the food items have been identified. Our research shows that our suggested method not only guarantees precision but also has the potential to greatly streamline the complex, time-consuming, and labor-intensive processes now used for estimating calorie intake by turning them together into real-time automation techniques.

**Keywords**- deep learning; convolution neural networks; image processing; food calorie; object recognition system;

## I. INTRODUCTION

There has been a dramatic increase in the prevalence of obesity in recent decades, making it a major issue for public health. Obesity is associated with an increased risk of several life-threatening diseases and conditions, such as diabetes, cardiovascular disease, high cholesterol, some forms of cancer (especially breast and colon cancer), and breathing problems [1]. The obesity pandemic is mostly attributable to poor dietary choices. Bad eating habits include things like often nibbling between meals and consuming plenty of sugary foods. Obesity is defined as a body mass index (BMI) of 30 or above [2]. Maintaining a healthy body mass index (BMI) necessitates adhering to guidelines for how many calories a person should consume each day. Thus, fighting obesity requires eating well-balanced meals that are low in both fat and calories. Therefore, it is essential to have accurate ways of forecasting and monitoring one's daily calorie consumption. An enormous benefit in this regard is that the meal itself may provide a rough estimate of the calorie content. However, to the best of our knowledge, no piece of medical technology exists that can instantly ascertain the number of calories contained in a given food item. In the food industry, it is common practice to include each ingredient and its calorie count on the manufacturer's label. McDonald's, one of the best fast-food chains, for example, provides calorie counts for each item on the menu [3]. This tagging is done manually using a calorie table endorsed by health experts [4]. Although it may help one restrict their calorie consumption, the process is expensive, time-consuming, prone to mistakes, and, most importantly, not very effective. It's possible that developing a system to calculate the number of calories in meals instantaneously using image recognition might be a workable solution to this problem. This technique might allow for faster and more accurate calorie counting. To provide some perspective, a computer vision system trained on a controlled data set can distinguish between different kinds of food in images. The occurrence of alterations in pictures produced by different situations [5] makes it challenging to create a food picture-based classification system. Among them include shifts in illumination, variations in available foods, and physical obstructions. Therefore, it is crucial to consider an adequate set of parameters while developing pattern recognition systems [6], given the limitations of supervised learning.

Calorie content may be estimated from a photograph [9,10], although most research in this area has focused on identifying the meals seen in photographs [7,8]. According to an examination of the claimed effects, most of the approaches require a significant investment of time and energy [11]. It's also tough to utilize them for a dinner with multiple items since most image recognition algorithms weren't intended to calculate food calories. Therefore, it is helpful to have a compact, time- and space-efficient method of detecting many objects simultaneously for use in determining meal calories. Calorie predictions from food photographs may be made with reasonable accuracy using Convolution Neural Networks (CNN) [13].

This study develops an automated calorie estimation system using convolutional neural networks (CNNs) to enhance previous methods. Since this system is meant to run on a computing phone with an embedded camera, it is very easy to recognize food products to determine the constituent calories by relying on a specified given dataset of daily calorie intake. The development of a real-time picture recognition method has been accomplished. We can respond instantly to the user's query, often in a matter of milliseconds. This system can identify the kind of food in an input image (apples, bananas, mango, doughnuts, etc.), calculate its volume, and check the results against the current nutritional data table using image processing and segmentation. Characteristics that help in segmentation include texture, color, shape, and item size, each of which contributes to an object's recognizability. Briefly summarized are the most important takeaways from this investigation:

- Training and optimizing the prototype to achieve an accuracy of 85%;
- Conducting a comparison study among different settings of the CNN-based methodology in terms of accuracy, speed, and complexity;
- Creating a parametric compact CNN model to instinctually evaluate food images and approximate component calories by sensing distinguishable products in it.

## II. Related Works

The studies that have been conducted on the topic of picture classification and calorie estimate are thoroughly discussed in the review of relevant literature. Thus, the suggested models' performances are evaluated across five dimensions, including "actual time," "optimized time complexity," "optimal space complexity," and the "satisfying score." Generally speaking, a system's performance is considered good if the score is above 80% accurate. Table 1 documents the assessment's executive summary and highlights the study's unique contribution to previous research on this topic.

Using Multiple Kernel Learning and image characteristics including Gabor features, color histograms, bags of features (BoFs), and gradient histograms, Hoashi et al. offer an automatic food picture identification system for 85 food categories [14]. (MKL). Image categorization is addressed; however, calorie calculation is beyond the scope of this study. Pouladzadeh et al. [15] offer a support vector machine-based method for measuring the caloric and nutritional content of food (SVM). Their method makes use of image processing techniques applied to food and the data presented in a nutrition table. Issues are graded by the system, which is installed on mobile devices. Liang and Li examine a special data collection of food images in [16], which includes information about the meals' weights and dimensions. Foods are identified and total calorie counts are estimated using a deep learning method (Faster R-CNN). a total of 2978 images make up their data collection. A major drawback of this method is that it does not take into account contextual factors when estimating calories. The caloric content of food is estimated by Raikwar et al. using just photographs as input [17]. Several image processing methods are done to the food picture before the SVM is used. The author does not, however, discuss the estimation's real-time nature.

You just look once, quicker area convolutional neural network and only one multibox detector are only a few of the modern object recognition techniques discussed by Menezes et al. in [18]. However, the authors don't pay much attention to the accuracy of estimating calories when eating. The authors of [13,19-21] use a deep learning (DL)-based algorithm to estimate the number of calories in food photographs. Nevertheless, these models are laborious and cannot be used for the real-time estimate. Calorie counting using DL models is the subject of other research. As an example, Kasyap et al. [22] utilize a DL model to measure the calories in meals while reducing the inaccuracy by 20%. In [23], Ayon et al. use an innovative DL model on the website photos to provide instantaneous predictions about the number of calories in different foods. Okamoto et al. [24] use a similar strategy by searching the web for photographs of food and then preprocessing them to train a DL model to estimate food calories.

While similar work is in development, it is not yet scalable or real-time and can only provide an estimate of the caloric value of a meal based on a picture of the recipe [25]. In [26], Naomi et al. utilize HoloLens to make accurate calorie and portion size estimates in a short amount of time.

Jelodar and Sun build a pipeline [27] to calculate caloric intake and replicate meals in varying sizes. However, because they solely care about accuracy, their solution is quite expensive both in terms of processing and scalability. Reconstructing the 3D form of the food and plates from a single photograph is the focus of Naritomi and Yanai's work [28], where they introduce the notion of hungry networks. Since rendering 3D pictures take a considerable amount of time, this approach lengthens the processing time. Suburban et al. [29] propose a combination of the Mask R-CNN and GrabCut methods, which take around three minutes to calculate, to increase the precision of segmentation operations and calorie computation. In [30], Siemon et al. aim to achieve the same thing using a more precise transfer learning strategy based on hierarchical clustering. On the other hand, their approach needs food clustering data to be available ahead of time and increases computational complexity. Last but not least, Zaman et al. employ a 3D volume estimate of food photos and associated nutrient volume estimation [31], but this method needs specialized hardware to execute and is hence unsuitable for practical usage.

Taken together, these points suggest that current practices do not meet any of the five criteria listed in Table 1. This work uses the opportunity to address this knowledge gap by creating a portable, CNN- based, real-time calorie estimate system for meals. This technique is also applicable to the widespread usage of smart gadgets.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 9***
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

TABLE I. COMPARATIVE ANALYSIS OF FOOD CALORIE SYSTEMS

| Studies | Year | Food Calorie Estimation | Real-Time | Optimize Time Complexity | Optimize Space Complexity | Satisfactory Score |
|---|---|---|---|---|---|---|
| Hoashi et al. [14] | 2010 | - | - | - | - | ✓ |
| Pouladzade et al. [15] | 2014 | ✓ | - | ✓ | ✓ | - |
| Liang & Li [16] | 2017 | ✓ | - | - | - | ✓ |
| Raikwar et al. [17] | 2018 | ✓ | - | ✓ | ✓ | - |
| Meneze et al. [18] | 2019 | - | ✓ | - | - | ✓ |
| Zaman et al. [31] | 2019 | - | - | ✓ | ✓ | - |
| Poply et al. [13] | 2020 | ✓ | - | - | - | ✓ |
| Latif et al. [19] | 2020 | ✓ | - | - | - | ✓ |
| Shen et al. [20] | 2020 | ✓ | - | - | - | ✓ |
| Ruede et al. [25] | 2020 | ✓ | - | - | - | ✓ |
| Kasyap et al. [22] | 2021 | ✓ | - | - | - | - |
| Ayon et al. [23] | 2021 | ✓ | - | - | - | - |
| Darapaneni et al. [21] | 2021 | ✓ | - | - | - | ✓ |
| Okamoto et al. [24] | 2021 | ✓ | - | - | - | - |
| Naritomi et al. [26] | 2021 | ✓ | - | - | - | - |
| Jelodar & Sun [27] | 2021 | ✓ | - | - | - | - |
| Naritomi & Yanai [28] | 2021 | ✓ | ✓ | - | - | ✓ |
| Siemon et al. [30] | 2021 | - | - | - | - | ✓ |
| Suburban [29] | 2022 | ✓ | - | - | - | ✓ |
| Proposed system | 2023 | ✓ | ✓ | ✓ | ✓ | ✓ |

## III. AIM AND OBJECTIVES

### A. *Plan Going on a Real-time basis*

When dealing with a real-time system, responses must occur within the allotted window of time. There are two main ways to categorize real-time systems: hard real-time systems and soft real-time systems. In the former, time constraints must be satisfied without fail, but in the latter, there is a small chance that the time restriction will be missed on rare occasions [32]. This research suggests a soft real-time system.

### B. *Convolutional neural with Learning Techniques*

Artificial neural networks (ANNs) based on deep learning are widely used for visual image classification in a classification data set [33]. Given that CNN is not a completely linked network, it requires less processing power [34]. Due to this quality, CNN is preferable to other methods when trying to classify images [35]. The following are the layers that make up a traditional CNN model.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 9***
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

### *C. Convolution Layer*

Information about a picture is saved in the computer as a matrix, with each pixel's value kept intact. At this stage, many filters are in operation. A filter is a matrix as well; however, it's much smaller than the matrix used to create a picture. Every filter dimension in a convolution layer is the same, but the values might vary. Input images are scanned by the filters, the values of the image's matrix are multiplied by the filter's matrix, and the result is a new matrix.

### *D. Maxpolling Layer*

In many neural network architectures, the max-pooling phase is employed after every convolution layer. This max-pooling layer's primary responsibility is to do feature extraction. It looks at the matrix produced by the convolution layer and takes the most essential characteristic from it. For the deep learning model, this means substantially more efficiency.

### *E. Dense Layer*

When it comes to connectivity, the thick layer has it all. In a dense layer, each neuron or filter is linked to every node in the layer below it in the network's output. It's a little classical neural network built within the CNN, to be exact [36]. Each neuron in this layer receives one of the outputs from the layer below it, and all of the neurons in this layer pass their outputs onto the next layer.

### *F. Rectified Linear Unit (ReLU) Activation*

Without affecting the convolution layer's receptive fields, this activation function enhances the network's decision and nonlinear properties. Since ReLU can train the neural network multiple times quicker than other nonlinear models used during CNNs (such as nonlinear function, a relative of a nonlinear function, and sigmoid), it is generally favored.

### *G. Adam Optimizer*

Adam, an optimization approach based on stochastic gradient descent, may be used instead of the more common stochastic gradient descent method to iteratively update network weights in light of training data [37]. It stores a reduced average of prior gradients $m(t)$, as well as a lowered average of past squared gradients, including such AdaDelta and RMSprop.

$$m(t) = \beta_1 m(t-1) + (1-\beta_1)\delta w(t) \tag{1}$$

$$v(t) = \beta_2 v(t-1) + (1-\beta_2)\delta w(t) \tag{2}$$

### *H. SoftMax Function*

With this operation, a matrix of $K$ real values is transformed into a matrix of $K$ actual values that add up to $1$. SoftMax transforms the input values, which may be optimistic, minus, zero, or perhaps more than one, into probabilistic values ranging from 0 and 1.

$$\sigma(\vec{Z})_i = \frac{e^{Z_i}}{\sum_{j=1}^{K} e^{Z_j}} \tag{3}$$

$Z_i$ values are members of the input vector, and they may be anything in the real number range. The normalizing factor ensures that the total of all the values returned by the function is always one, as shown in the last part of the formula.

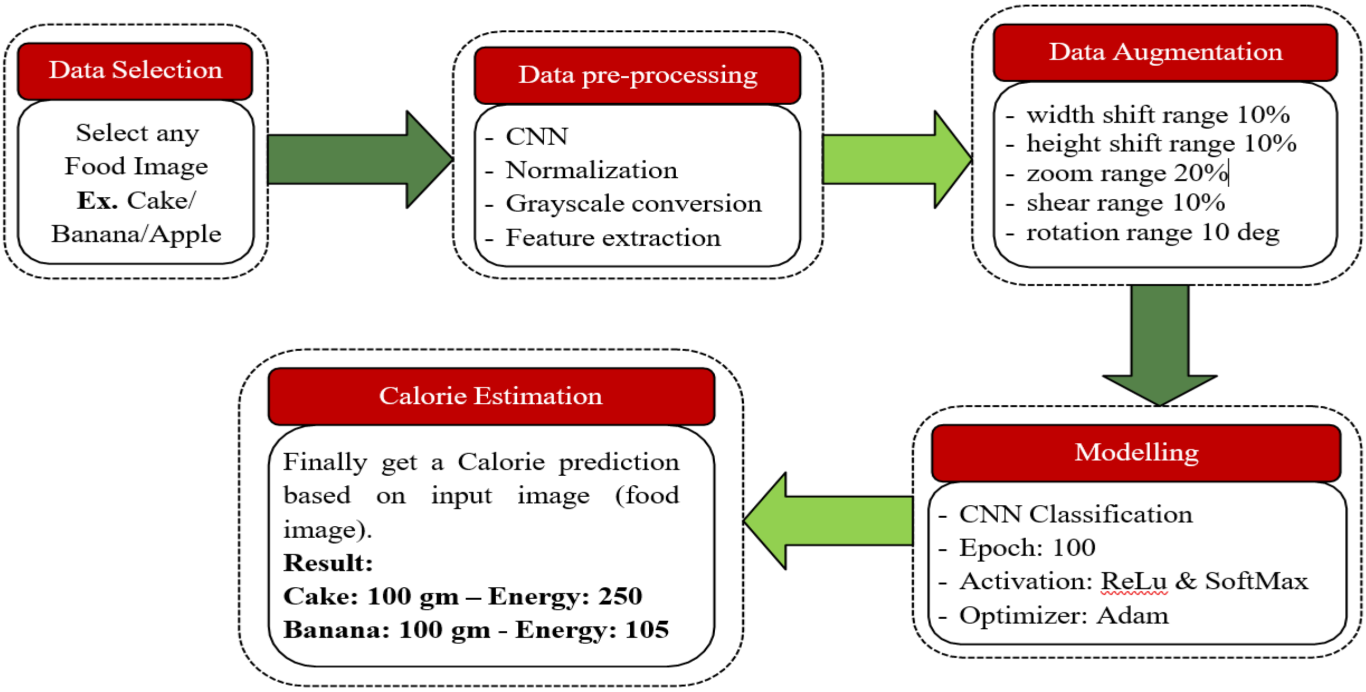


Figure 1. System model for image-based food calorie estimation

## IV. METHODS

Data set identification, source data, which was before, upsampling, and model creation are the activities that materialize this study. The steps of our technique are shown in Figure 1 above.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 9***
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

### *A. Selections of datasets*

The purpose of this research is to do classification using qualitative data collection. Five different kinds of food are represented in the dataset. Since the dataset is symmetric, there is an equal number of instances of each meal category. To produce a more precise outcome, two sets of information were selected from Kaggle. Food-101 [38] and Fruit-360 [39] are the names of the two databases in question. The RGB photos of food products are included in these data sets. There are a thousand pictures in each group. Detailed top-down and side-scans of each food group were saved, as were photos from throughout the food chain. For the goal of calculating calories, each dataset is accompanied by an implied food caloric list and meal volume. Parameter examples for the data set are shown in Table 2.

TABLE II. DATA SELECTION FROM TWO DIFFERENT DATA SET WITH A TYPICAL NUTRITION TABLE [4].

| Source | Data Set Name | Types | No. of Instances | Volume (Gram) | Energy (Kilocalorie) |
|---|---|---|---|---|---|
| Kaggle | FOOD-101 and Fruit-360 | Apple | 1000 | 133 | 72 |
| | | Banana | | 118 | 105 |
| | | Donut | | 64 | 269 |
| | | Cupcake | | 72 | 262 |
| | | Mango | | 133 | 68 |

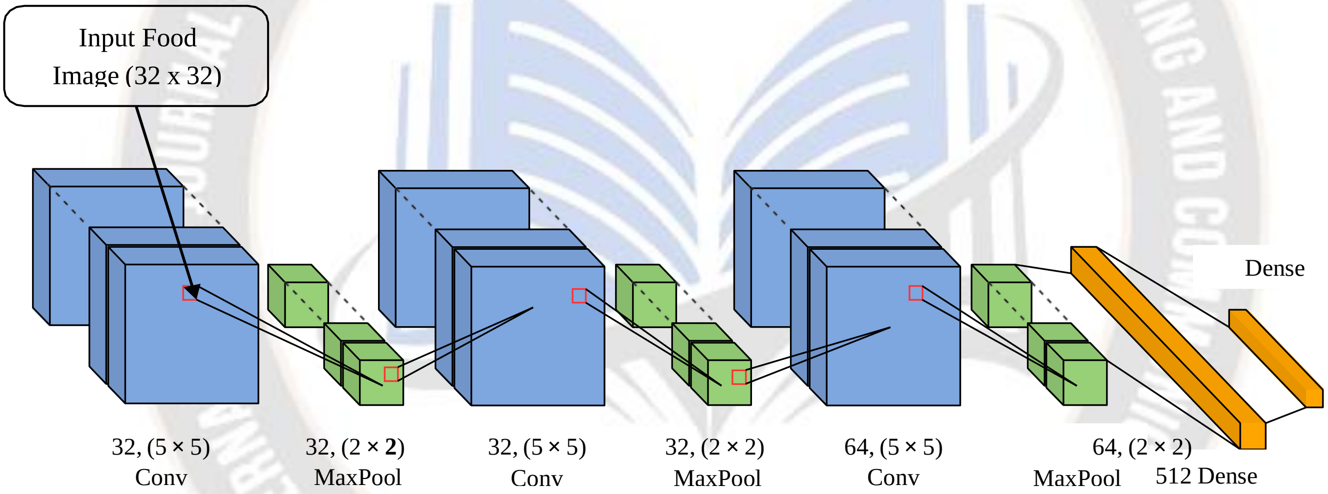


Figure 2. The architecture of the proposed CNN

### *B. Preprocessing of Datasets*

The purpose of this procedure is mostly logistical; the final resolution of the photos in the dataset is 32 x 32 pixels. Following that, the dataset underwent picture normalization based on the images' RGB values. Image normalization guarantees the most accurate evaluation across all data collection methods and texture examples. Consequently, the Color information channel is split into 255 values in this investigation, and the photographs in the dataset are converted to grayscale. In the end, this reduces the disparity between the RGB image of the associated pictures. After converting the picture to monochrome, the statistical method of feature extraction was used. The histogram for an image is a distribution of grayscale values that reveals how often each value occurs in the picture. For the histogram to work, it must be assumed that the gray values of the forefront (anatomy components) and background (beyond the patient border) are discernible. It also modifies an image's overall contrast by recalibrating the intensity of each pixel.

### *C. Intelligence Enhancement*

Adding slightly altered copies of current data or producing novel artificial information based on existing data is one example of data augmentation. As a regularized, it reduces the likelihood of the overfitting issue occurring during the development of a machine-learning model. Accidental removal of some remaining variables (i.e., noise) represented in the therapy incorporates structure is known as overfitting [40]. Data augmentation serves the same function in our investigation. To supplement the data, we utilized TensorFlow's picture data generator feature. The function is categorized as an image function in the Keras subtype of TensorFlow [41]. The heightened parameters are shown in Table 3. In this research, the training portion accounts for 80%, the validation portion for 10%, and the testing portion accounts for 10%.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 9***
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

TABLE III. AUGMENTATION PARAMETERS

| Parameters | Values |
|---|---|
| Width shift range (%) | 10 |
| Height shift range (%) | 10 |
| Zoom range (%) | 20 |
| Shear range (%) | 10 |
| Rotation range (deg) | 10 |

### *D. Building a Model*

The process of identifying the optimal model set up for a certain data collection is time-consuming and complex. This research has constructed a generic model by optimizing its parameters for a specific data set. Thus, 81 completely customized models for the new CNN technique were produced by this research. A schematic of the CNN model is depicted in Figure 2.

To create the CNN model, this research employs a variety of finely calibrated parameters, including sampling frequency, filtering variety, pooling size, and packed nodes. In addition, activation functions like conv2D layer and relu are employed. Model 44 of the analyzed 81 CNN models had the highest accuracy, and this model will be further detailed below.

Execution times and other model characteristics for the top 10 models are shown in Section 5. There is currently no computer that can accurately determine the number of calories in any given meal, and there is also no which was before-food calorie picture collection that can be used to train a model.

## V. RESULTS OUTCOMES

In this part, we create the performance assessment matrix and explain the model's performance in terms of metrics like appropriate moments and model space complexity. A model's inference time is the total amount of time needed to run all of its operations.

$$inference\ time = \frac{\text{FLOPs}}{\text{FLOPs}} \tag{4}$$

**FLOPs:** We have determined the total amount of calculations executed by a model as a proxy for its inference time. The phrase "Floating Point Operation" (FLOP) is introduced here. Any mathematical operation on a floating-point number is included here. How complicated a model is measured in FLOPs.

$$\begin{aligned} Convolutions - FLOPs \\ &= 2 \times Number\ of\ Kernal \\ &\times Kernal\ Shape\ \times Output\ Shape \end{aligned} \tag{5}$$

$$\begin{aligned} Fully\ Connected\ Layers - FLOPs \\ &= 2 \times Input\ Size\ \times Output\ Size \end{aligned} \tag{6}$$

**FLOPS:** The FP/s metric is the following one (FLOPS). The effectiveness of the equipment system is described here. For the sake of this analysis, 1 FLOPS is equal to 1 billion operations per second.

The calculation of space complexity is crucial for an actual food calorie estimating system. This equation represents the practical realization of the optimization problem of a CNN model.

$$CNN\ model\ space\ complexity = (cwhk + k) \times p \tag{7}$$

where $c$, $w$, $h$, and $k$ denote the width, height, and quantity of output networks, kernels, and channels, respectively. The byte count of each element is denoted by the symbol $p$. In this analysis, we focus on a per-element size of 4 bytes (floating point).

A model's performance is guaranteed to improve if its accuracy is both greater during training and lower during validation. In contrast, the error rate is assessed by zeroing in on the optimal model-specific hyperparameter. A total of 80 training epochs were used on all models. To enhance the performance of models, tinkering of modeling techniques and prototype parameters are employed. The model is tuned using the following filter parameters: number of filters (16), size of filters (3), size of filters (5), and size of filters (7). Filtering is often used to clean up picture data by getting rid of unwanted elements like noise and artifacts. To avoid the overfitting issue, we employ the drop (0.5) function and model-oriented parameters like the pool size (2,2) for feature extraction and the dense node (512) for image comparison. To keep the feature map's probability from being disrupted, activation functions like ReLu as well as SoftMax are used. A program called Adam is used to find the optimal configuration for the data.

TABLE IV. GENERAL CNN MODEL STRUCTURE

| Groups | Layers |
|---|---|
| Group 1 (tunable) | Conv2D, Conv2D, and MaxPooling2D |
| Group 2 (tunable) | Conv2D, Conv2D, and MaxPooling2D |
| Group 3 (tunable) | Conv2D, Conv2D, and MaxPooling2D |
| Group 4 (tunable) | DropOut, Flatten, Dense Layer, and DropOut |

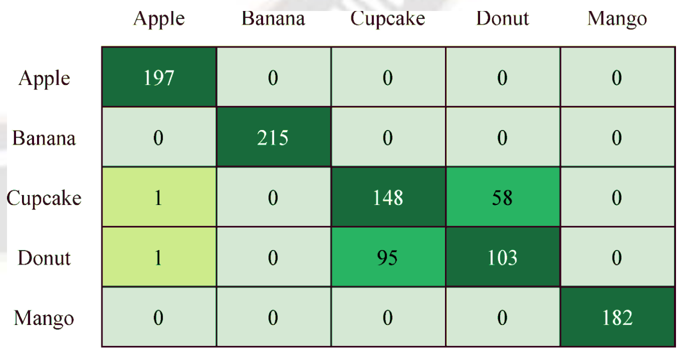


Figure 3. Confusion Matrix

Table 4 displays the 81 models that were created and how they were categorized into four distinct sets. For the most part, the models fell short of the mark. There is a nearly 80% rate of correctness. Models with a filter size of (5,5), on the other hand, provide higher validation accuracy (3,3). Table 5 compares the top 10 models in great detail.

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 9***
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

TABLE V. TOP 10 MODELS PERFORMANCE COMPARISON FOR THE ACCURACY (A), LOSS (L), SPACE (IN BYTES), AND TIME (IN SECONDS).

| Model Name | Group-1 Filter Num | Group-2 Filter Num | Group-3 Filter Num | Filter Size | Training | Validation | Test | Space | Time(s) |
|---|---|---|---|---|---|---|---|---|---|
| model16 | 16 | 32 | 64 | (3,3) | A: 0.795 L: 0.38 | A: 0.847 L: 0.29 | A: 0.836 L: 0.3 | 30,500 | 0.0005 |
| model17 | 16 | 32 | 64 | (5,5) | A: 0.809 L: 0.34 | A: 0.847 L: 0.27 | A: 0.829 L: 0.28 | 66,340 | 0.0008 |
| model23 | 16 | 64 | 32 | (5,5) | A: 0.80 L: 0.37 | A: 0.86 L: 0.27 | A: 0.85 L: 0.27 | 66,340 | 0.0008 |
| model26 | 16 | 64 | 64 | (5,5) | A: 0.818 L: 0.34 | A: 0.85 L: 0.26 | A: 0.84 L: 0.27 | 82,340 | 0.0013 |
| model44 | 32 | 32 | 64 | (5,5) | A: 0.84 L: 0.31 | A: 0.86 L: 0.25 | A: 0.848 L: 0.26 | 74,340 | 0.0008 |
| model50 | 32 | 64 | 32 | (5,5) | A: 0.81 L: 0.36 | A: 0.85 L: 0.26 | A: 0.829 L: 0.28 | 74,340 | 0.0010 |
| model52 | 32 | 64 | 64 | (3,3) | A: 0.82 L: 0.32 | A: 0.82 L: 0.31 | A: 0.79 L: 0.34 | 39,140 | 0.0007 |
| model62 | 64 | 16 | 64 | (5,5) | A: 0.80 L: 0.35 | A: 0.84 L: 0.27 | A: 0.836 L: 0.28 | 82,340 | 0.0014 |
| model68 | 64 | 32 | 32 | (5,5) | A: 0.81 L: 0.35 | A: 0.84 L: 0.28 | A: 0.836 L: 0.29 | 74,340 | 0.0010 |
| model70 | 64 | 32 | 64 | (3,3) | A: 0.83 L: 0.34 | A: 0.836 L: 0.32 | A: 0.837 L: 0.31 | 39,140 | 0.0007 |

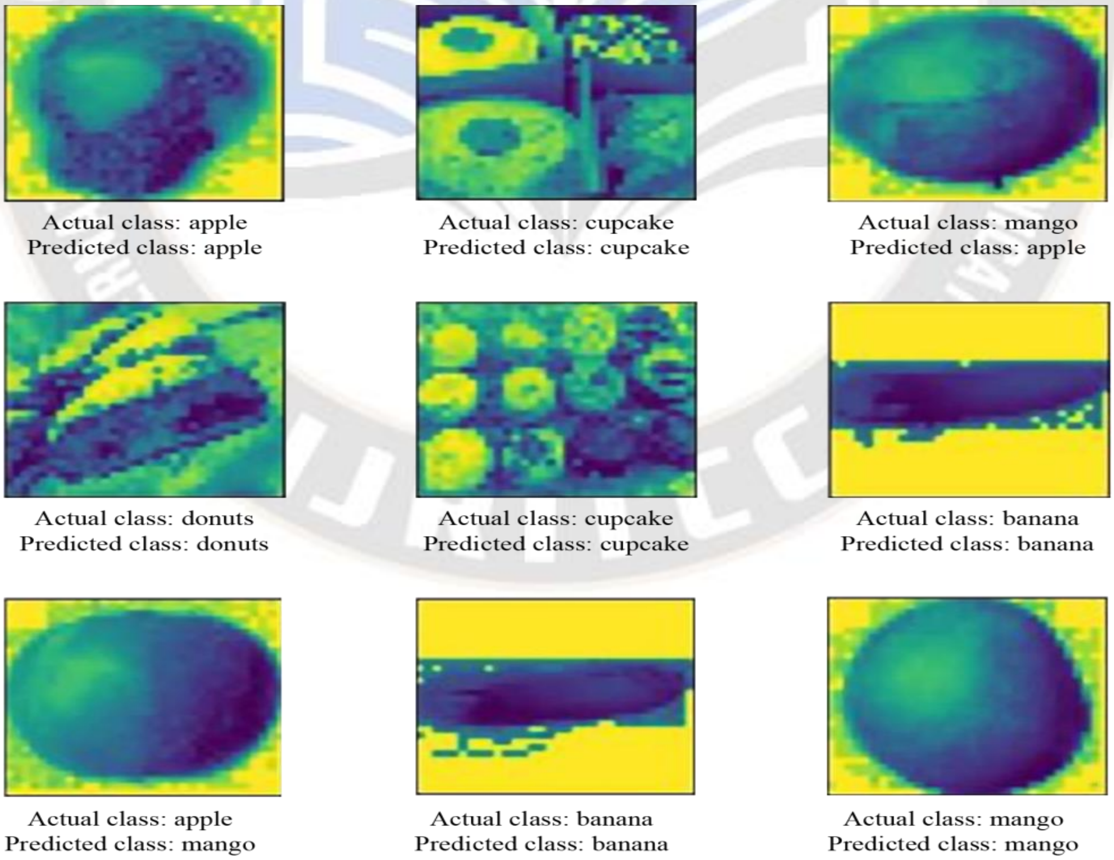


Figure 4. Predicted Labels

*International Journal on Recent and Innovation Trends in Computing and Communication*
***ISSN: 2321-8169 Volume: 11 Issue: 9***
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

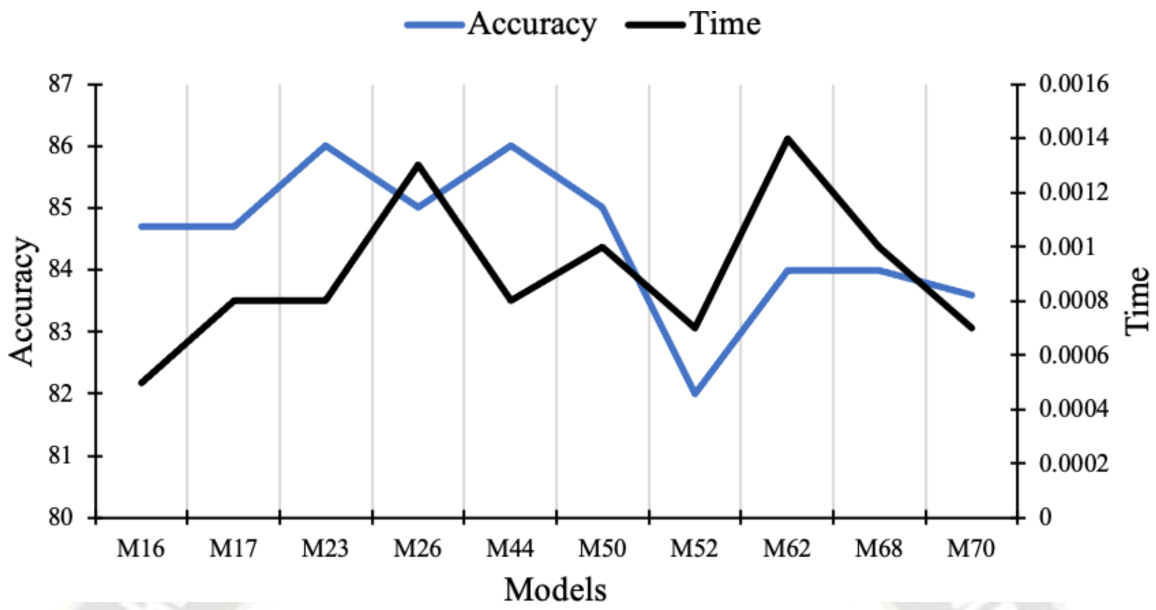


Figure 5. Accuracy for 10 models

The data in the table shows that when the control action is (5,5), the pool length is (2,2), and indeed the filtration number is (), model 44 produces the optimal outcome (32,32,64). Model 44 achieves the best validation, which includes the greatest accuracy in both training and testing. Model 44 has an 86% accuracy rate and an 84.9 % test accuracy. Model 44, as seen in the graph, exhibits a validation loss of 25% and a test loss of 26%, both of which are lower than the losses shown for the other models. As for training loss, it's 31%, while accuracy is 84%. Images from both the expected and actual food classes are shown in Figure 3. Figure 4 is a line graph showing how each of the top 10 categories performs in terms of accuracy and runtime, with model 44 coming out on top.

To choose the optimal model in real-time, further analysis was undertaken based on three criteria: precision, efficiency, and speed. In Figure 5, we have a ternary diagram depicting this situation. Originally, precision, area, and duration were all scaled using a min-max algorithm. There will be no more than two decimal places in the final total. Both the weight and the velocity were determined by dividing the corresponding scaled values of space and time by 1. Model 17 has superior performance across all three parameters when compared using the Ternary diagrams. The optimum value in the tripartite graph is the one that is around the triangle's hypocenter. Model 17 is the most appropriate model for estimating food calories since it is located towards the centers of the ternary diagrams.

## VI. Results and Discussions

To accomplish the study's aims, the authors presented a practical CNN model. The success of the model may be attributed to the strategic placement of hidden neurons inside its thick layers. Also, it has an adequate amount of discontinuities in its neural connections, which helps it avoid the overfitting issue. Our data set-specific CNN models make use of a wide variety of filter frequencies and filter sizes. Furthermore, it demonstrates that the optimal model changes depending on the viewpoint from which the observations are made. The greatest option here, in terms of both precision and efficiency, is model 44. Model 44 can handle 125 frames a second since it needs 0.008 seconds to process each one. Our approach may easily be implemented as mobile-based real-time apps, processing 60 frames per second even with added overheads. Once again, Model 17 is the optimal option if precision, efficiency, and convenience are all taken into account.

The technique was developed to help dietitians treat patients who are overweight or obese. Those who use the system will be able to exercise more command over their daily food intake, which will improve their health. But there is always a way to become better. This also holds for the suggested model. But for a deeper grasp, it's essential to train the system using photographs of a wide variety of foods, so that it can accurately recognize any kind of food. Due to not having access to high-quality picture data collection meeting the requirements, the scope of this research is limited to obtaining one feature. Using a laptop camera, the current technology is capable of real-time data analysis. The goal of this study is to eventually make the system accessible via a wide range of smart portable devices. At now, bespoke data sets are used for calorie calculation of food photos. Feature extraction is also essential for improving the training and validation results of an image identification system. However, the suggested models did not perform better than 90% accuracy. A future goal is to improve the accuracy of the food picture recognition system's training and validation by optimizing the feature extraction method. In addition, there is a strategy to adjust for the varying amounts of food to arrive at the most precise calorie calculation possible.

## VII. Conclusion

To successfully moderate one's eating habits, automated food picture detection and matching nutrient content estimation with the highest accuracy are crucial. In this study, we create a compact, optimized CNN model by trying out several different configurations and achieving an accuracy of around 85%. Simple linear processes may be used to train the algorithm and then apply it to specific data sets for improved accuracy. The method may help solve a social problem by facilitating the maintenance of a diet plan for people of varying body mass indexes, including those who are obese. However, there are plans to perform more precise research in the field of food picture recognition and calorie estimate.

## REFERENCES

[1] Bray, G.; Bouchard, C. (Eds.) Handbook of Obesity-Volume 2: Clinical Applications; CRC Press: Boca Raton, FL, USA, 2014.
[2] Prentice, A.M.; Jebb, S.A. Beyond body mass index. Obes. Rev. 2001, 2, 141–147. [CrossRef]
[3] Petimar, J.; Ramirez, M.; Rifas-Shiman, S.L.; Linakis, S.; Mullen, J.; Roberto, C.A.; Block, J.P. Evaluation of the impact of calorie labeling on McDonald's restaurant menus: A natural experiment. Int. J. Behav. Nutr. Phys. Act. 2019, 16, 99. [CrossRef]
[4] Health Canada. Health Canada Nutrient Values. November 2011. Available online: https://www.canada.ca/en/health-canada/services/food-nutrition/healthy-eating/nutrient-data/nutrient-value-some-common-foods-booklet.html (accessed on 31 August 2022).
[5] Kasar, M.M.; Bhattacharyya, D.; Kim, T.H. Face recognition using neural network: A review. Int. J. Secure. Its Appl. 2016, 10, 81–100. [CrossRef]
[6] Li, G.Z.; Bu, H.L.; Yang, M.Q.; Zeng, X.Q.; Yang, J.Y. Selecting subsets of newly extracted features from PCA and PLS in microarray data analysis. BMC Genom. 2008, 9, S24. [CrossRef]
[7] Ciocca, G.; Micali, G.; Napoletano, P. State recognition of food images using deep features. IEEE Access 2020, 8, 32003–32017. [CrossRef]
[8] Park, S.J.; Palvanov, A.; Lee, C.H.; Jeong, N.; Cho, Y.I.; Lee, H.J. The development of food image detection and recognition model of Korean food for mobile dietary management. Nutr. Res. Pract. 2019, 13, 521–528. [CrossRef]
[9] Mezgec, S.; Seljak, B.K. Using deep learning for food and beverage image recognition. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019; pp. 5149–5151.
[10] Mezgec, S.; Eftimov, T.; Bucher, T.; Seljak, B.K. Mixed deep learning and natural language processing method for fake-food image recognition and standardization to help automated dietary assessment. Public Health Nutr. 2019, 22, 1193–1202. [CrossRef]
[11] Moolch, ani, D.; Kumar, A.; Sarangi, S.R. Accelerating CNN inference on asics: A survey. J. Syst. Archit. 2021, 113, 101887. [CrossRef]
[12] Liang, H.; Gao, Y.; Sun, Y.; Sun, X. CEP: Calories estimation from food photos. Int. J. Comput. Appl. 2020, 42, 569–577. [CrossRef]
[13] Poply, P. An Instance Segmentation approach to Food Calorie Estimation using Mask R-CNN. In Proceedings of the 2020 3rd International Conference on Signal Processing and Machine Learning, Beijing, China, 22–24 October 2020; pp. 73–78.
[14] Hoashi, H.; Joutou, T.; Yanai, K. Image recognition of 85 food categories by feature fusion. In Proceedings of Proceedings of the 2010 IEEE International Symposium on Multimedia, Taichung, Taiwan, 13–15 December 2010; pp. 296–301.
[15] Pouladzadeh, P.; Shirmohammadi, S.; Al-Maghrabi, R. Measuring calorie and nutrition from food image. IEEE Trans. Instrum. Meas. 2014, 63, 1947–1956. [CrossRef]
[16] Liang, Y.; Li, J. Computer vision-based food calorie estimation: Data set, method, and experiment. arXiv 2017, arXiv:1705.07632.
[17] Raikwar, H.; Jain, H.; Baghel, A. Calorie Estimation from Fast Food Images Using Support Vector Machine. Int. J. Future Revolut. Comput. Sci. Commun. Eng. 2018, 4, 98–102.
[18] De Menezes, R.S.T.; Magalhaes, R.M.; Maia, H. Object recognition using convolutional neural networks. In Recent Trends in Artificial Neural Networks-from Training to Prediction; IntechOpen: London, UK, 2019.
[19] Latif, G.; Alsalem, B.; Mubarky, W.; Mohammad, N.; Alghazo, J. Automatic Fruits Calories Estimation through Convolutional Neural Networks. In Proceedings of the 2020 6th International Conference on Computer and Technology Applications, Antalya, Turkey, 14–16 April 2020; pp. 17–21.
[20] Shen, Z.; Shehzad, A.; Chen, S.; Sun, H.; Liu, J. Machine learning based approach on food recognition and nutrition estimation. Procedia Comput. Sci. 2020, 174, 448–453. [CrossRef]
[21] Darapaneni, N.; Singh, V.; Tarkar, Y.S.; Kataria, S.; Bansal, N.; Kharade, A.; Paduri, A.R. Food Image Recognition and Calorie Prediction. In Proceedings of the 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Toronto, ON, Canada, 21–24 April 2021; pp. 1–6.
[22] Kasyap, V.B.; Jayapandian, N. Food Calorie Estimation using Convolutional Neural Network. In Proceedings of the 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 13–14 May 2021; pp. 666–670.
[23] Ayon, S.A.; Mashrafi, C.Z.; Yousuf, A.B.; Hossain, F.; Hossain, M.I. FoodieCal: A Convolutional Neural Network Based Food Detection and Calorie Estimation System. In Proceedings of the 2021 National Computing Colleges Conference (NCCC), Taif, Saudi Arabia, 27–28 March 2021; pp. 1–6.
[24] Okamoto, K.; Adachi, K.; Yanai, K. Region-Based Food Calorie Estimation for Multiple-Dish Meals. In Proceedings of the 13th International Workshop on Multimedia for Cooking and Eating Activities, Taipei, Taiwan, 16–19 November 2021; pp. 17–24.
[25] Ruede, R.; Heusser, V.; Frank, L.; Roitberg, A.; Haurilet, M.; Stiefelhagen, R. Multi-task learning for calorie prediction on a novel large-scale recipe data set enriched with nutritional information. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 4001–4008.
[26] Naritomi, S.; Yanai, K. CalorieCaptorGlass: Food calorie estimation based on actual size using hololens and deep learning. In Proceedings of the 2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW), Atlanta, GA, USA, 22–26 March 2020; pp. 818–819.
[27] Jelodar, A.B.; Sun, Y. Calorie Aware Automatic Meal Kit Generation from an Image. arXiv 2021, arXiv:2112.09839.
[28] Naritomi, S.; Yanai, K. Pop'n Food: 3D Food Model Estimation System from a Single Image. In Proceedings of the 2021 IEEE 4th International Conference on Multimedia Information Processing and Retrieval (MIPR), Tokyo, Japan, 8–10 September 2021; pp. 223–226.
[29] Subaran, T.L.; Semiawan, T.; Syakrani, N. Mask R-CNN and GrabCut Algorithm for an Image- based Calorie Estimation System. J. Inf. Syst. Eng. Bus. Intell. 2022, 8, 1–10. [CrossRef]
[30] Siemon, M.S.; Shihavuddin, A.S.M.; Ravn-Haren, G. Sequential transfer learning based on hierarchical clustering for improved performance in deep learning based food segmentation. Sci. Rep. 2021, 11, 813. [CrossRef]
[31] Zaman, D.M.S.; Maruf, M.H.; Rahman, M.A.; Ferdousy, J.; Shihavuddin, A.S.M. Food Depth Estimation Using Low-Cost Mobile-Based System for Real-Time Dietary Assessment. GUB J. Sci. Eng. 2019, 6, 1–11. [CrossRef]
[32] Buttazzo, G.; Lipari, G.; Abeni, L.; Caccamo, M. Soft Real-Time Systems; Springer: Berlin/Heidelberg, Germany, 2005; Volume 283.
[33] Heenaye-Mamode Khan, M.; Boodoo-Jahangeer, N.; Dullull, W.; Nathire, S.; Gao, X.; Sinha, G.R.; Nagwanshi, K.K. Multi-class classification of breast cancer abnormalities using Deep Convolutional Neural Network (CNN). PLoS ONE 2021, 16, e0256500. [CrossRef]
[34] Jaiswal, S.; Nandi, G.C. Robust real-time emotion detection system using CNN architecture. Neural Comput. Appl. 2020, 32, 11253–11262. [CrossRef]
[35] Albawi, S.; Mohammed, T.A.; Al-Zawi, S. Understanding of a convolutional neural network. In Proceedings of the 2017 international conference on engineering and technology (ICET), Antalya, Turkey, 21–23 August 2017; pp. 1–6.

[36] Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.

[37] Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. arXiv 2014, arXiv:1412.6980.

[38] Kaggle Data Set: Food-101. Available online: https://www.kaggle.com/datasets/dansbecker/food- 101 (accessed on 1 July 2022).

[39] Kaggle Data Set: Fruit-360. Available online: https://www.kaggle.com/datasets/moltean/fruits (accessed on 1 July 2022).

[40] Jabbar, H.; Khan, R.Z. Methods to avoid over-fitting and under-fitting in supervised machine learning (comparative study). Comput. Sci. Commun. Instrum. Devices 2015, 70, 163–172.

[41] TensorFlow v2.10.0. 2021. Available online: https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/image/ImageDataGenerator (accessed on 26 April 2021).