

A Project Report
on
FOOD CALORIE PREDICTION USING
Machine learning

Submitted By:
Akshay kumar(20103010)

Bachelor of Technology in Computer Science and Engineering



ABSTRACT

Now a days, people are having very busy schedules due to changes in their lifestyle and work commitments. People are not concentrating in their food habits, hence it leads to obesity. Obesity is becoming a common problem in today's modern life. So that we need a system which can make changes to the food choices of people and provides them with instruction that leads to effective results of maintaining their body. If a system informs the nutritional information of a food item and classify it as healthy or non-healthy to the user, then people are able to identify their daily intake of calorie value of their food items.

Food is the Major need of Any Human Body, Without food a human body cannot exist or survive. In the current scenario the lifestyle of modern life is changing day by day and with these changes the Composition of Human Body need also changes which include several consumptions or ready made foods. With the excess of foods their is one issue which rises in our daily life are Obescity , Fat and these issue are generated due to over access of calorie to Human Body . In this Paper we aimed to prepare a solution of this problem. Our Work is to first determine the Category of food and after predicting the category of Food(Fruit or Vegetable) Our System determining the Category of that Image(Either the Image is in the Category of Food or Vegetable) After Determining these Things We are predicting the Calorie of that Food as well. In This Paper Our Work is totally based on the Algorithm named as MobileNet. MobileNet is the Conjunction or the Deep learning and Transfer Learning that is used to recognize the image and based on The image it will Identify the Category. Our System Comprises with several segmentation and image parameters as well.

KEYWORDS: MobileNet , Deep Learning , Transfer Learning , Streamlit , CNN.

Table of Contents

Title	Page
Candidates Declaration	No.
Acknowledgement	
Abstract	
List of Table	
List of Figures	
Acronyms	
Introduction	1
1.1 Introduction	
1.2 Formulation of Problem	6
1.2.1 Tool and Technology Used	9
Functionality/Working of Project	18
Results and Discussion	27
Conclusion and Future Scope	50
5.1 Conclusion	53
5.2 Future Scope	55
Reference	60

CHAPTER - 01

1.1 INTRODUCTION

When people's Body Mass Index (BMI) is over thirty (kg/m^2), they're usually thought-about to be corpulent. High BMI will increase the chance of diseases like cardiopathy [1]. the most reason of fat is attributable to the imbalance between the number of caloric intake (consumption) and energy output (expenditure). due to disposition to record and track, lack of connected nutritionary info or alternative reasons, patients typically expertise hassle in dominant the number of calories they consume. There square measure countless projected strategies to estimate calories supported laptop vision [2, 3, 4, 5], however once the authors' analysis, the accuracy of detection and volume estimation still ought to be improved. during this paper, the most distinction from alternative similar approaches is that it needs associate input of 2 pictures, and also the use quicker R-CNN to sight the item and GrabCut algorithmic rule to get every food's contour. After that, the authors will estimate every food's volume and calories.

To survive in the world human Body needs food to make a living and great practice diet plan. In the current Days innovations and enhancement allows people and humans to judge their balanced diet and it also increases to become a good healthy and fit body and these things are only possible when a human observing their nutritional status. Many Research institutes have already implemented several machine Learning models and visions that may help human body to monitor their balanced diet status.

Healthy eating, which includes counting calories and vitamins, is an increasing number of turning into greater essential for most consumers anywhere inside the global. consistent with a have a look at organized through Canadian obesity community ,approx one in fivemen or women and one in

10 young ones in the World are actually residing with several health related issues like obesity. This ensures that approx of six millionsof peoples are tormented by these problems, some

distance greater than humans being with sugar, coronary heart failure, persistent lung sickness, or several types of cancer as well. Comparative patterns already exist all around the earth prompting calls for planning individual assistive frameworks to assist shoppers with dealing with their eating.

We have got visible an expensive way of fitness co-related Phone packages. Many of those Phone softwares routinely report physical sports that helps to count the energy that is burnt while doing any work or exercise in each day. A number of calorie count apps count the calorie that is intake by a human body in each day. Last Year One survey is happen that ensures that cellular calories intake consumption size, that indicates professionals and various pros and cons associated with that work, is published in North America in the year 2016. There are several other applications as well that are worked on both on Mobiles and Laptops. It gives a way for human being to count the calorie burnt by the body during the exercises. Gadgets plays a very important role for doing these types of activities.



Examples of single-item foods.

whilst the recent system labored are good as compared to exist one, just like many other systems that are preceding paintings, it supposed that the intake meals picture contains simplest one or more food items, as proven above. In different phrases, the simplest work is to use single item meals but

for single-items meals it is not possible that the human can take only single item meals. There may be a chance that there is combination of food items in a dish as well so there may be a failure in using single item food. In this Paper, we are going to prepare a deep learning system that would help human to know how much calorie should a human body have to take so that they can make their body fit. Many such models have opted various gadgets that serve as a way to estimate the calories. The purpose of proposed realizing system vision is to reduce the selection of objects, thereby reducing the burden on users. Additionally, using of simple machine learning module named as OpenCV in the service system can also estimate and identify the way of objects in the moving process from human body to the electronic gadgets. In theory, that might speed up the system and this research paper aims to illustrate an algorithm that is not very heavily loaded and provides excellent accuracy. To achieve this, we will use the concepts of transfer learning and MobileNet. There are large systems that were introduced to predict the calorie but the system that we are proposing are based on two algorithms that are CNN and MobileNet. MobileNet is one of the major algorithms that is used for recognizing the images and also it is very light weighted as well. For the Proposed System MobileNet works on two different ways. First it is used to detect the food category and after that it helps to detect the category of fruits and vegetables as well. For all this Transfer Learning also plays a very important role as well.

Transfer learning is one of major versatile and efficient deep learning method in which the model acts as a two way source. In transfer learning generally the output of first model is taken as input of second model as so on. Common example of transfer learning in the form of deep learning is when we need to predict our own predictive models then transfer deep learning plays a very significant role. MobileNet-v2 is a 53-layer deep convolutional neural network. You can see the web training version

from ImageNet data. The retrained network can convert images into 1000 objectcategories, such as keyboard, mouse, mouse, many animals.

MobileNet Architecture

We used MobileNet architecture presented by Howard et al. [6] that is designed and based on depthwise separable convolutions to build a lightweight deep CNN that makes a model too small and reduces the computation time. illustrates the MobileNet architecture. Consequently, MobileNet can be implemented for several recognition problems such as object detection, face attributes, fine-grain classification, and landmark recognition.

Proposed MobileNet Architecture

Our proposed MobileNet architecture was as follows. First, we used the pre-trained model of MobileNet architecture. We decided to remove three layers, including the average pooling, fully connected, and softmax layers from the original network. Second, three extra layers; the global average pooling (GAP) layers, the batch normalization (BN), and softmax layers are attached. The main objective of our proposed MobileNet architecture is helping the network to train faster and achieving higher accuracy. Then, the dropout method is proposed to prevent overfitting. Also, the batch normalization layer helps the network to train faster. The activation function called the rectified linear unit (ReLU) is computed between the batch normalization layer and the dropout layer. After we applied the GAP layers instead of the average pooling, it shows that the parameters in the model are decreased, and impact directly on the size of the model. Finally, for training the proposed network, we used the fine-tuned MobileNet to train the network on the ETH Food-101 dataset

Data Augmentation Techniques

Data augmentation is a technique to generate new training image data that relate to the same image. Many data augmentation techniques such as rotation, horizontal, vertical, flip, width shift, height shift techniques are applied to the image recognition problems and the accuracy performance is improved [22]. Samples of image augmentation are shown in figure 4. In this paper, the data augmentation techniques applied to our experiments consists of rescaling, rotation, width shift, height shift, horizontal flip, shear, and zoom.

Existing System

There are some applications are present in this user's food calorie management. But each one has its own drawbacks. The following are the available applications in the existing system.

Fatsecret

Fatsecret application is Australia based application. In this application user need to add a capture of their food items and add tag name to that then it will give calorie information of the food item. User can share what they eat to others and connect with their friends. This application mainly gives the diet plans to the user to reduce their weight. But to use the diet plans, they need to buy a premium membership for some cost. They give barcode scanner for packed food items to detect and give nutritional information of food items. The bar code scanner does not work with Indian packed foods. In this they provide connect with our dietitian by invite them but it also require premium membership. It gives the report of what they ate and calorie information of their intake. Overall this application is good but works well only for foreign people. It doesn't provide diet plans and correct nutrition values of all Indian foods. In this application, the developer did not use deep learning or any other technologies to classify food items.

Nutrition plus

Nutrition Plus is an application where the user can search for the food items based on the nutrition values such as carbohydrates, protein, fat, iron, calcium and vitamin. The user can also search by

food items and find the particular food's nutrient composition. It gives the healthcare information about the diseases and provides the diet plans. They didn't use any technologies for the classification of food images

Calorie Mama

Calorie mama is another application in this domain. It provides calorie and other nutritional information of food items by capturing the food items. It uses image classification technology to classify food items. It works well with foreign foods and some of Indian foods. It gives foreign meal plans to gain, reduce; maintain body weights for their premium membership holders. This is a paid application after free trails, the user need to upgrade the account. This application will not suggest Indian foods to user based on their health conditions.

Motivation

Computer vision has been introduced to estimate calories from food images. But current food image datasets don't contain volume and mass records of foods, which leads to an incomplete calorie estimation. Current obesity treatment techniques require the patient to record all food intakes per day. In most of the cases, unfortunately patients have troubles in estimating the amount of food intake because of the self-denial of the problem, lack of nutritional information, the manual process of writing down this information (which is tiresome and can be forgotten), and other reasons.

1.2

FORMULATION OF PROBLEM

Calorie Estimation. within the present merchandise of proprietary, calorie the visit is usually completed in individual through searching on the information ingredients and meals costs. there may be aa big range of automated food tracking strategies using mobile packages consisting of . but,these require tons more consumer interaction along with high quantity of mannual input. Its act as an instance of the images,where users are acquired to classify sizes once more times even when the ingredients are not much good themselves, which makes this fun most of the time and mistakes prone.on this work, we are

talking about the feature of imaginative and prescient-primarily based food evaluation, that is intended to offer facts on supported nutrients in container images. lots of imaginative and prescient-based sports for food description(overview of methods and data in compliance with multi or binary-level procedures.

For those images, the photograph is separated by ixel intelligence food and non-food providers, observed picture separation in a hard and fast set of classes, along with cuisines or meals tire. the subsequent segment carries extent and quantity weight dimension, and vitamins facts about the step. vitamins statistics is decided based totally on extraction features from re-skilled CNNs are restrained to segments produced inside the first segment. as an example, Myers et al. divided via the given photograph, producing food categories, components, and corresponding extent.

Calories they are then given credit score in comparison to the restrained extent as soon as the density of the food enterprise versus nutritious meals database. in addition, the metadata is similar to the GRSS vicinity as properly user vitamins signs have been used to enhance recommendations. even though arroaces have many classes increase a sure basis methods of measuring satisfactory have a negative range. First, these methods require an outline and category of every segment, wherein every sub-phase separates. exclusive phases are tough to remedy appropriately. similarly, those fashions encompass a step-by means of-step class of rixels into meals and non-food classes as nicely degree the volume of a meal. So, one will need sensible ixel announcements of excessive value information as nicely more records about food in the picture as its volume. in addition, arroaces with more than one categories require a description of the distinctive categories and contributions and outcomes of the complete stage, to a lesser extent. This ends in fact useful records from preceding sections isn't always dispersed once more will no longer be used in later sections to enhance pointers

Meals is an important and basic need of human life and it is the first priority of the many healthcare centres. In our day to day life many different dietary assessment are designed to make people understand

their every day ingesting behavior ,exploring nutrients constraints and help to prepare a very healthy diet plan.

As based on the popularity of meals we designed three step set of rules to study and analysed food item from day by day meals pics. This set of rules are advised to acknowledge multi item images with the aid of detecting the food items and its category and using convolutional neural network (CNN) for object category . Using vicinity suggestion network(RPN) is also implemented from faster –CNN version, the gadget first generates a couple of location of proposals on enter photographs .It then identifies each entered food item by mapping them into characteristics map which categorizes them into distinct food classes ,additionally the gadget will calculate and analysed the dietary component calorie,fat,protein and carbohydrates .Within the evaluation using that would work with proposed system approximately food gadgets supported FOOD101 with boundations and limitations. There are many version that were tested /evaluated by different assignments and the practical consequences show that our machine ensures to acknowledge meals objects appropriately and it generates the dietary evaluation document effectively so the user can access it to enhance frame fitness and health.

in this discipline, numerous reputation strategies and methods of identification have been in short defined. The problem assertion is the identical in all strategies, distinctive parameters are selected, specific records and many others., in distinct fashions also are described. the inducement for the outsourced work to isolate the Indian snakes and shield the development app better than the redesigned work is revealed. stepped forward model merchandise also are mentioned which may be the future characteristic of a damaged device. version guide is expected to be about ninety five% which would be the high-quality access to covered service to this point the use of the DCNN parcel with the aid of imparting a limited verification guarantee. on this diagram, we located a manner to differentiate and become aware of high-colored snakes (together with burgers, pizzaetc.)

In a survey photo to measure the houses in them. In our trial we use CNN in PFID statistics which offers ninety four% of ninety four% better than BOF. And the false fine degree isn't so excessive. nowadays they are very worried about their health. consequently, along side a ill character, a health care worker with first rate dietary capability can benefit from this product. inside the future, we are able to try to improve the provider by way of developing a sturdy device so as to accurately discover all types of snakes This company offers meals testing that identifies task improvement improvements. The advanced version separates the nutrients in rlate with a department based on CSWWLIFC. The CSW-WLIFC-department department uses the SSW-primarily based work component inside the separation gadget. first rate, e.g., refinement, sharing, and overhead have been eliminated from numerous sections to include a vector facts framework. WLMNN Gweightlosser orders the basic components that guide meals from the food plan and eliminates all the nutrients. improved food frame layout has reduced the check ratings, as an example, MA, MSE, and SA. nutrients material used for recycling from UNIMIB2016 records.

The improvement of an stepped forward food testing framework is documented with contemporary fashions, as an instance, WLI-FC for LM-NN, SSSW-WLIFC for LMNN, and WLIN-FC and WLM. effects recreationdemonstratethat le modelproposed with CSWS-WLIFC based division. Paper presents a place to begin for interest in preferred meal strategies and advanced statistics on these strategies. superior experiments primarily based on a diffusion of dietary variation strategies and the extraordinary country wide size structures cited in this area. As upkeep facts raises extra focus of sources related to resources to acquire those centered recommendations are evolved, changed and furnished to a exceptional degree. An electronic machine, for example, cleverphone provides a limited amount of access weight reduction facts for document staff and much less than five aid courses.

From their survey got here extraordinary techniques and programs that manipulate the diverse models. however there is an extension of the frameworks that is straightforward to utilise and includes all age on a variety of foundations. moreover you want to make responses to get a combined profits for mixed diets

Convolutional neural Network achieved ample improved than did outdated approaches By means of handcrafted features. Complete comment of competent convolution kernels, we inveterate that color structures are vital to nutrition image identification . Defined nutrition identification consuming a minor dataset, which was proposed to be secondhand in a Smartphone based food classification scheme . Which identifies unhealthy foods beginning cartridges of eating and guess meal calories created on identifying diets Estimating the ideal heaviness to trust diverse image features with MKL, they take attained the 61.34% classification percentage for 50 types of diets through the cross-validation-based assessment accuracy of 62% whereas SVM came up with an accuracy of 67% and our CNN model built with TensorFlow gave an accuracy of 97% . Also, we show here how the model behaves with hyper parameter tuning by changing parameters like the learning rate and number of neurons in each hidden layer which govern how effectively a model behaves. The working model if this tensor flow is divided into different layer. That is sub category of L1 and L2 that will be helpful for optimizing the task. Tensorflow's Object detection API to detect multiple food items in each image and then using mathematical calculations to find the calorie content of food classes present in the image There are multiple flows of below approach such as Thumb, Simple Food Images, and Size assessment. So we utilize thumb for the volume expectation utilizing clients, the thumb is anything but a proficient way. Since other than the enlisted client whoever utilizes this than the precision will drop. Just Works on Simple Food Images, this methodology doesn't chip away at complex food varieties like soup, sandwich and so forth it just deals with straightforward food sources like apple, banana and so on . I utilized a dataset of inexpensive food pictures dependent on the Pittsburgh Fast-Food Image Dataset. Mathworks image processing toolbox is utilized for separating highlights . Absolute of 11,868 crude highlights separated from RGB portrayal of the picture. Crude highlights decreased utilizing Principal Component Analysis (PCA) and Information Gain (InfoGain) to 23 highlights . Utilizing these highlights and Sequential Minimal Optimization (SMO) they arranged the food. Size expectation is finished by utilizing Random Forest in grams. At long last, the calorie of the food is anticipated utilizing multilayer

perceptron. They investigate various sorts of portrayal of a picture, for example, Averaged RGB, Gray Scale, BW 0.7 and BW 0.5 however they get the best outcomes with RGB portrayal . For the size assessment, the ground truth estimation of the food will be taken from its producer. On account of that this technique gives a low precision on the food sources from different producers. White Background, they utilized a white foundation for the food sources in this tasks dataset. Therefore, this venture isn't effective for day by day. A dataset of 2978 pictures of 19 distinctive food were taken. So I utilized Faster RCNN for object discovery . Each jumping box that made by Faster R-CNN is arranged. Utilizing the GrabCut calculation for picture division I separated food into 3 classes' ellipsoid, segment and unpredictable . Utilizing the coin as a kind of perspective point we can ascertain the volume of the food relying upon its shape. Realizing the volume mass can be handily determined utilizing the thickness of the food. With the mass is realized we can figure the calorie of the food.

1.2.1

TOOLS AND TECHNOLOGY USED

DATASETS AND METHODS

Datasets

For our paper, annotated images of the Indian meal were required. For this we extracted custom dataset from images.google.com and labelled the images using labelImg. Our custom dataset contains 500 annotated images of 6 classes and split 2/3 and 1/3 in train and validation dataset. 6 classes of the dataset being Bhaji(vegetable), Dal(curry), Rice, Roti, Puri and Gulab Jamun.

CNN (custom dataset)

We performed binary classification on custom dataset of the Indian Roti which helped us predict a given image to be a Roti or not. In this we used a total of 200 Roti and non-roti images.

YOLO V2 (COCO dataset)

We used the COCO dataset to train the YOLO model first. It contains 80 object classes such as person, laptop, apple, etc. With 35000 images in the dataset with train, valid, test split in 1/2,1/4,1/4 respectively. Annotations being in XML and text format

YOLO V3 (custom dataset)

Our custom dataset contains 500 annotated images of 6 classes and split 2/3 and 1/3 in train and validation dataset. 6 classes of the dataset being Sabji (vegetable), Dal (Indian curry), Rice, Roti, Puri and Gulab Jamun.

Methods

The first step in our project is to register a plate and a reference object which will be a 1-rupee or 2-rupee or 5-rupee coin and by using OpenCV modules we will calculate the dimensions of the plate that will be further used to give the calorie count of the food item. For the food detection we are using YOLO (You Only Look Once) which is a deep learning algorithm. The dataset will be labelled and trained according to the YOLO format and after the food detection the counted. For all the items except for rice we will be using standard calorie values and for rice we will calculate the area of it and eventually we will get the calorie count through mathematical computations. The formula for the area calculation that we are using is:

$Fr = Sr * (Fp / Sp)$ where,

- Fr: area of target food item.
- Fp: the area of the registered plate.
- Sp: the pixel count of the whole registered plate.
- Sr: the pixel count of the region of the target food item.

After getting the calorie counts of each item the aggregated calorie count of the meal will be presented to the user.

MATERIAL AND STRATEGIES

Deep Learning

Deep learning is a component of a broader conception of machine learning ways supported learning knowledge and its representations. Learning are often carried in 3 ways supervised, semi-supervised or unattended. Deep learning consists of following architectures like deep neural ne process, audio recognition, social network filtering, computational linguistics, bio-informatics, medical image analysis, material scrutiny and parlor game programs, wherever they need made results love and in some cases superior to human specialists.

Deep Learning based mostly Objection Detection

The authors selected quicker R-CNN rather than victimisation linguistics segmentation technique like absolutely Convolution Networks (FCN). Here, once the pictures square measure inputted as RGB channels, the authors will get a series of bounding boxes, which implies the category if judge

FUNCTIONALITY/ WORKING OF PROJECT

The project consists of two steps, identifying food from an image and converting the food identified into a calorie estimation. We performed food image classification using CNN (convolutional Neural Network). Steps followed:

Pre-processing:

Some basic pre-processing has been performed to clean the dataset where the irrelevant and noisy images of 15 categories have been removed.

Also, data augmentation has been performed –

- Pixel values re-scaled in the range of [0,1].
- Random rotations max 40 degree.
- Random zoom applied
- Shear angle in counter-clockwise direction in degrees

Trained the model:

We trained the model with images of 15 categories using the classifier CNN (convolutional Neural Network) which is a class of deep, feed forward artificial neural networks that has successfully been applied to analyzing visual imagery.

Convolutional Neural Network

The convolutional Neural Network (CNN) offers a technique for many general image classification problems. It has been applied in food classification and resulted in a good accuracy. CNN is widely used in food recognition and provides high performance than the

traditional methods. Over the last few years, due to the enhancements in the deep learning, especially in the convolutional neural networks, the accuracy in detecting and recognizing food images has been increased. This is not only because larger datasets but also new algorithms and improved deep architectures. convolutional Neural Network (CNN) is also known as LeNet due to its inventor. CNN mainly comprises convolutional layers, pooling layers and sub-

sampling layers followed by fully-connected layers. The CNN takes an input image and applies convolutional and then sub-sampling. After two such computations, the data is fed into the fully connected neural network, where it performs the classification task. The main advantage of CNN is the ability to learn the high-level efficient features and in addition to that, it is robust against small rotations and shifts.

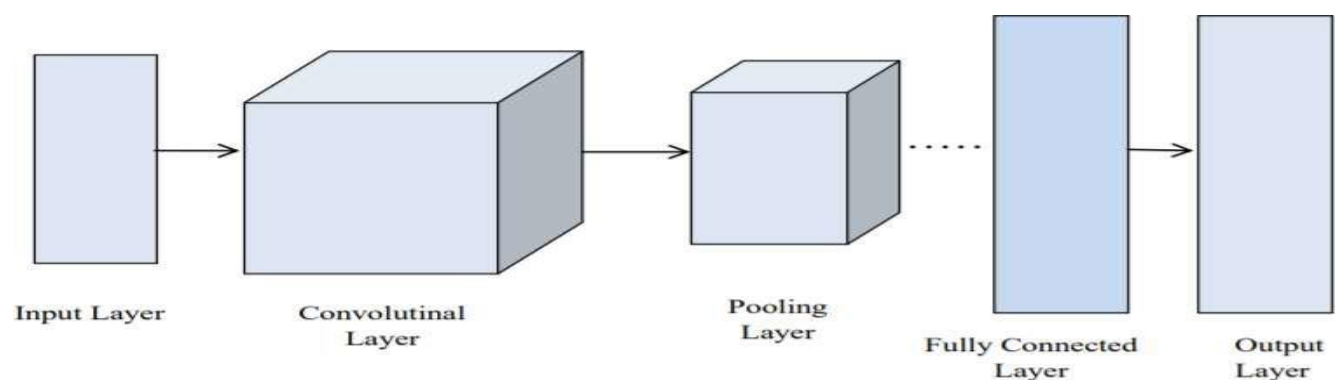


Image classification!

The convolutional neural network (CNN) is a class of deep learning neural networks. CNNs represent a huge breakthrough in image recognition. They're most commonly used to analyze visual imagery and are frequently working behind the scenes in image classification. They can be found at the core of everything from Facebook's photo tagging to self-driving cars. They're working hard behind the scenes in everything from healthcare to security.

They're fast and they're efficient. But how do they work?

Image classification is the process of taking an **input** (like a picture) and outputting a **class** (like "cat") or a **probability** that the input is a particular class ("there's a 90% probability that this input is a cat"). You can look at a picture and know that you're looking at a terrible shot of your own face, but how can a computer learn to do that?

With a convolutional neural network!

- Convolutional layers
- ReLU layer
- Pooling layers
- a Fully connected layer

A classic CNN architecture would look something like this:

**Input ->Convolution ->ReLU ->Convolution ->ReLU ->Pooling -> ReLU
->Convolution ->ReLU ->Pooling ->Fully Connected**

A CNN **convolves** (not convolutes...) learned features with input data and uses 2D convolutional layers. This means that this type of network is ideal for processing 2D images. Compared to other image classification algorithms, CNNs actually use very little preprocessing. This means that they can **learn** the filters that have to be hand- made in other algorithms. CNNs can be used in tons of applications from image and video recognition, image classification, and recommender systems to natural language processing and medical image analysis.

CNNs are inspired by biological processes. They're based on some [cool research done by Hubel and Wiesel in the 60s](#) regarding vision in cats and monkeys. The pattern of connectivity in a CNN comes from their research regarding the organization of the visual cortex. In a mammal's eye, individual neurons respond to visual stimuli only in the receptive field, which is a restricted region. The receptive fields of

different regions partially overlap so that the entire field of vision is covered. This is the way that a CNN works.

CNNs have an input layer, and output layer, and hidden layers. The hidden layers usually consist of convolutional layers, ReLU layers, pooling layers, and fully connected layers.

- Convolutional layers apply a convolution operation to the input. This passes the information on to the next layer.
- Pooling combines the outputs of clusters of neurons into a single neuron in the next layer.
- Fully connected layers connect every neuron in one layer to every neuron in the next layer.

In a convolutional layer, neurons only receive input from a subarea of the previous layer. In a fully connected layer, each neuron receives input from *every* element of the previous layer.

A CNN works by extracting features from images. This eliminates the need for manual feature extraction. The features are not trained! They're learned while the network trains on a set of images. This makes deep learning models extremely accurate for computer vision tasks. CNNs learn feature detection through tens or hundreds of hidden layers. Each layer increases the complexity of the learned features.

A CNN

- starts with an input image
- applies many different filters to it to create a feature map
- applies a ReLU function to increase non-linearity
- applies a pooling layer to each feature map
- flattens the pooled images into one long vector.
- inputs the vector into a fully connected artificial neural network.

- processes the features through the network. The final fully connected layer provides the “voting” of the classes that we’re after.
- trains through forward propagation and backpropagation for many, many epochs. This repeats until we have a well-defined neural network with trained weights and feature detectors.

So what does that mean?

At the very beginning of this process, an input image is broken down into pixels. Based on that information, the computer can begin to work on the data.

For a color image, this is a 3D array with a blue layer, a green layer, and a red layer. Each one of those colors has its own value between 0 and 255. The color can be found by combining the values in each of the three layers.

What are the basic building blocks of a CNN?

Convolution

The main purpose of the convolution step is to extract features from the input image. The convolutional layer is always the first step in a CNN.

You have an input image, a feature detector, and a feature map. You take the filter and apply it pixel block by pixel block to the input image. You do this through the multiplication of the matrices.

The light from the flashlight here is your **filter** and the region you’re sliding over is the **receptive field**. The light sliding across the receptive fields is your flashlight **convolving**. Your filter is an array of numbers (also called weights or parameters). The distance the light from your flashlight slides as it travels (are you moving your filter over one row of bubbles at a time? Two?) is called the **stride**. For example, a stride of one means that you’re moving your filter over one pixel at a time. The convention is a stride of two.

The depth of the filter has to be the same as the depth of the input, so if we were looking at a color image, the depth would be 3. That makes the dimensions of this filter 5x5x3. In each position, the filter multiplies the values in the filter with the original values in the pixel. This is element wise multiplication. The multiplications are summed up, creating a single number. If you started at the top left corner of your bubble wrap, this number is representative of the top left corner. Now you move your filter to the next position and repeat the process all around the bubble wrap. The array you end up with is called a **feature map** or an **activation map**! You can use more than one filter, which will do a better job of preserving spatial relationships.

Transfer Learning:

Transfer learning means taking the relevant parts of a pre-trained machine learning model and applying it to a new but similar problem. This will usually be the core information for the model to function, with new aspects added to the model to solve a specific task.

Transfer Learning for Image Recognition

A range of high-performing models have been developed for image classification and demonstrated on the annual ImageNet Large Scale Visual Recognition Challenge, or ILSVRC.

This challenge, often referred to simply as [ImageNet](#), given the source of the image used in the competition, has resulted in a number of innovations in the architecture and training of convolutional neural networks. In addition, many of the models used in the competitions have been released under a permissive license. These models can be used as the basis for transfer learning in computer vision applications.

This is desirable for a number of reasons, not least:

- **Useful Learned Features:** The models have learned how to detect generic features from photographs, given that they were trained on more than 1,000,000 images for 1,000 categories.

- **State-of-the-Art Performance:** The models achieved state of the art performance and remain effective on the specific image recognition task for which they were developed.
- **Easily Accessible:** The model weights are provided as free downloadable files and many libraries provide convenient APIs to download and use the models directly.

The model weights can be downloaded and used in the same model architecture using a range of different deep learning libraries, including Keras.

How to Use Pre-Trained Models

The use of a pre-trained model is limited only by your creativity.

For example, a model may be downloaded and used as-is, such as embedded into an application and used to classify new photographs.

Alternately, models may be downloaded and use as feature extraction models. Here, the output of the model from a layer prior to the output layer of the model is used as input to a new classifier model.

Recall that convolutional layers closer to the input layer of the model learn low-level features such as lines, that layers in the middle of the layer learn complex abstract features that combine the lower level features extracted from the input, and layers closer to the output interpret the extracted features in the context of a classification task.

Armed with this understanding, a level of detail for feature extraction from an existing pre-trained model can be chosen. For example, if a new task is quite different from classifying objects in photographs (e.g. different to ImageNet), then perhaps the output of the pre-trained model after the few layers would be appropriate. If a new task is quite similar to the task of classifying objects in photographs, then perhaps the output from layers much deeper in the model can be used, or even the output of the fully connected layer prior to the output layer can be used.

The pre-trained model can be used as a separate feature extraction program, in which case input can be pre-processed by the model or portion of the model to a given an output (e.g. vector of numbers) for each input image, that can then use as input when training a new model.

Alternately, the pre-trained model or desired portion of the model can be integrated directly into a new neural network model.

In this usage, the weights of the pre-traine can be frozen so that they are not updated as the new model is trained. Alternately, the weights may be updated during the training of the new model, perhaps with a lower learning rate, allowing the pre-trained model to act like a weight initialization scheme when training the new model.

We can summarize some of these usage patterns as follows:

- **Classifier:** The pre-trained model is used directly to classify new images.
- **Standalone Feature Extractor:** The pre-trained model, or some portion of the model, is used to pre-process images and extract relevant features.
- **Integrated Feature Extractor:** The pre-trained model, or some portion of the model, is integrated into a new model, but layers of the pre-trained model are frozen during training.
- **Weight Initialization:** The pre-trained model, or some portion of the model, is integrated into a new model, and the layers of the pre-trained model are trained in concert with the new model.

Each approach can be effective and save significant time in developing and training a deep convolutional neural network model. It may not be clear as to which usage of the pre-trained model may yield the best results on your new computer vision task, therefore some experimentation may be required.

Models for Transfer Learning

There are perhaps a dozen or more top-performing models for image recognition that can be downloaded and used as the basis for image recognition and related computer vision tasks.

Perhaps three of the more popular models are as follows:

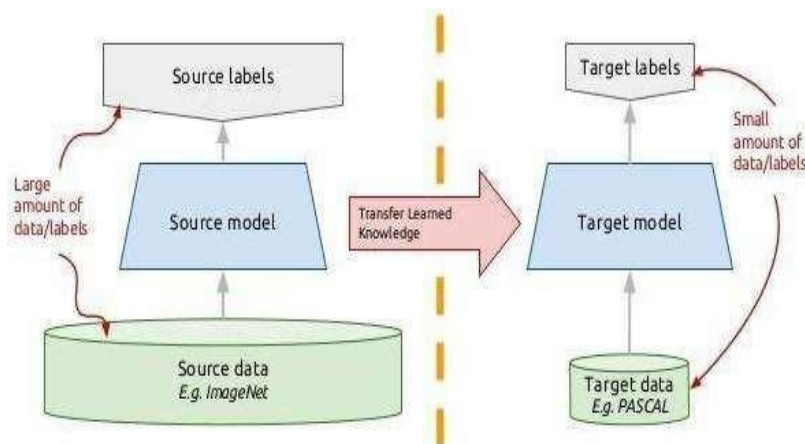
- VGG (e.g. VGG16 or VGG19).
- GoogLeNet (e.g. InceptionV3).
- Residual Network (e.g. ResNet50).

These models are both widely used for transfer learning both because of their performance, but also because they were examples that introduced specific architectural innovations, namely consistent and repeating structures (VGG), inception modules (GoogLeNet), and residual modules (ResNet).

Keras provides access to a number of top-performing pre-trained models that were developed for image recognition tasks. They are available via the [Applications API](#), and include functions to load a model with or without the pre-trained weights, and prepare data in a way that a given model may expect (e.g. scaling of size and pixel values).

The first time a pre-trained model is loaded, Keras will download the required model weights, which may take some time given the speed of your internet connection. Weights are stored in the `.keras/models/` directory under your home directory and will be loaded from this location the next time that they are used. When loading a given model, the “`include_top`” argument can be set to `False`, in which case the fully-connected output layers of the model used to make predictions is not loaded, allowing a new output layer to be added and trained.

Transfer learning: idea



Following is the general outline for transfer learning for object recognition:

- Load in a pre-trained CNN model trained on a large dataset
- Freeze parameters (weights) in model's lower convolutional layers
- Add custom classifier with several layers of trainable parameters to model
- Train classifier layers on training data available for task
- Fine-tune hyperparameters and unfreeze more layers as needed

This approach has proven successful for a wide range of domains. It's a great tool to have in your arsenal and generally the first approach that should be tried when confronted with a new image recognition problem.

RESULT AND DISCUSSION

Source Code and Outputs:

import the necessary libraries. (Remember the TensorFlow version, because we need to use the same version into the local machine)

```
import numpy as np
import pandas as pd
from pathlib import Path
import os.path
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow.keras.preprocessing.image import load_img, img_to_array
print(tf.__version__)
```

2.6.0

Now we need to define the Train, test, and validation images.

```
# Create a list with the filepaths for training and testing
train_dir = Path('../input/fruit-and-vegetable-image-recognition/train')
train_filepaths = list(train_dir.glob(r'**/*.jpg'))
```

```
test_dir = Path('../input/fruit-and-vegetable-image-recognition/test')
test_filepaths = list(test_dir.glob(r'**/*.jpg'))
```

```
val_dir = Path('../input/fruit-and-vegetable-image-recognition/validation')
val_filepaths = list(test_dir.glob(r'**/*.jpg'))
```

Now we need to create the data frame for each image with its label, So this is the function for it.

```
def image_processing(filepath):  
    """ Create a DataFrame with the filepath and the labels of the pictures  
    """  
  
    labels = [str(filepath[i]).split("/")[-2] \  
              for i in range(len(filepath))]  
  
    filepath = pd.Series(filepath, name='Filepath').astype(str)  
    labels = pd.Series(labels, name='Label')  
  
    # Concatenate filepaths and Labels  
    df = pd.concat([filepath, labels], axis=1)  
  
    # Shuffle the DataFrame and reset index  
    df = df.sample(frac=1).reset_index(drop = True)  
  
    return df
```

Now we need to make a function call for Train, Test, and Val images. Let's check how many labels and images we have in the dataset.

```
train_df = image_processing(train_filepaths)
test_df = image_processing(test_filepaths)
val_df = image_processing(val_filepaths)
```

```
print('-- Training set --\n')
print(f'Number of pictures: {train_df.shape[0]}\n')
print(f'Number of different labels: {len(train_df.Label.unique())}\n')
print(f'Labels: {train_df.Label.unique()}')
```

-- Training set --

Number of pictures: 3193

Number of different labels: 36

Labels: ['grapes' 'pomegranate' 'banana' 'carrot' 'garlic' 'onion' 'pineapple' 'orange' 'capsicum' 'jalepeno' 'paprika' 'watermelon' 'raddish' 'lettuce' 'spinach' 'tomato' 'sweetpotato' 'cauliflower' 'bell pepper' 'peas' 'lemon' 'kiwi' 'chilli pepper' 'cabbage' 'turnip' 'eggplant' 'potato' 'soy beans' 'pear' 'mango' 'beetroot' 'sweetcorn' 'cucumber' 'corn' 'apple' 'ginger']

Now let's check the generated data frame.

```
train_df.head(5)
```

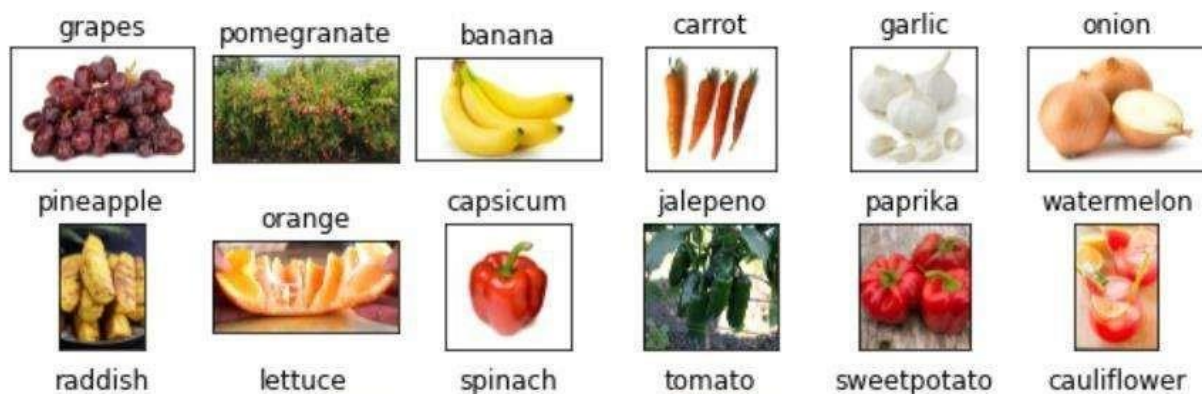
	Filepath	Label
0	../input/fruit-and-vegetable-image-recognition...	grapes
1	../input/fruit-and-vegetable-image-recognition...	pomegranate
2	../input/fruit-and-vegetable-image-recognition...	banana
3	../input/fruit-and-vegetable-image-recognition...	carrot
4	../input/fruit-and-vegetable-image-recognition...	garlic

Now let's check our labels of images are matching with the original image or not, We are going to use matplotlib to plot the images.

```
# Create a DataFrame with one Label of each category
df_unique = train_df.copy().drop_duplicates(subset=["Label"]).reset_index()

# Display some pictures of the dataset
fig, axes = plt.subplots(nrows=6, ncols=6, figsize=(8, 7),
                        subplot_kw={'xticks': [], 'yticks': []})

for i, ax in enumerate(axes.flat):
    ax.imshow(plt.imread(df_unique.Filepath[i]))
    ax.set_title(df_unique.Label[i], fontsize = 12)
plt.tight_layout(pad=0.5)
plt.show()
```



Now we need to generate the new images using these images, because we have a low number of images for each class, we are going to use the ImageDataGenerator module from the Keras, basically it will do the zoom, rotate, changing the color format, changing brightness and much more technique. With these techniques, it will generate new images.

```
train_generator = tf.keras.preprocessing.image.ImageDataGenerator(
    preprocessing_function=tf.keras.applications.mobilenet_v2.preprocess_input
)

test_generator = tf.keras.preprocessing.image.ImageDataGenerator(
    preprocessing_function=tf.keras.applications.mobilenet_v2.preprocess_input
)
```

Let's fit the images for training and testing.

```
train_images = train_generator.flow_from_dataframe(  
    dataframe=train_df,  
    x_col='Filepath',  
    y_col='Label',  
    target_size=(224, 224),  
    color_mode='rgb',  
    class_mode='categorical',  
    batch_size=32,  
    shuffle=True,  
    seed=0,  
    rotation_range=30,  
    zoom_range=0.15,  
    width_shift_range=0.2,  
    height_shift_range=0.2,  
    shear_range=0.15,  
    horizontal_flip=True,  
    fill_mode="nearest"  
)
```

Found 3193 validated image filenames belonging to 36 classes.

```
test_images = test_generator.flow_from_dataframe(  
    dataframe=test_df,  
    x_col='Filepath',  
    y_col='Label',  
    target_size=(224, 224),  
    color_mode='rgb',  
    class_mode='categorical',  
    batch_size=32,  
    shuffle=False  
)
```

Found 334 validated image filenames belonging to 36 classes.

Let's fit the pre-trained MobilenetV2 model.

```
pretrained_model = tf.keras.applications.MobileNetV2(
    input_shape=(224, 224, 3),
    include_top=False,
    weights='imagenet',
    pooling='avg'
)
pretrained_model.trainable = False
```

User settings:

```
KMP_AFFINITY=granularity=fine,verbose,compact,1,0
KMP_BLOCKTIME=0
KMP_SETTINGS=1
KMP_WARNINGS=0
```

Let's start training with our own images.

```
inputs = pretrained_model.input

x = tf.keras.layers.Dense(128, activation='relu')(pretrained_model.output)
x = tf.keras.layers.Dense(128, activation='relu')(x)

outputs = tf.keras.layers.Dense(36, activation='softmax')(x)

model = tf.keras.Model(inputs=inputs, outputs=outputs)

model.compile(
    optimizer='adam',
    loss='categorical_crossentropy',
    metrics=['accuracy']
)

history = model.fit(
    train_images,
    validation_data=val_images,
    batch_size = 32,
    epochs=5,
    callbacks=[
        tf.keras.callbacks.EarlyStopping(
            monitor='val_loss',
            patience=2,
            restore_best_weights=True
        )
    ]
)
```

```

Epoch 1/5
 2/100 [.....] - ETA: 2:33 - loss: 3.6501 - accuracy: 0.0469
/opt/conda/lib/python3.7/site-packages/PIL/Image.py:963: UserWarning: Palette images with Transparency expressed in bytes should be converted to RGBA images
  "Palette images with Transparency expressed in bytes should be converted to RGBA images"
100/100 [=====] - 226s 2s/step - loss: 1.9120 - accuracy: 0.4933 - val_loss: 0.4976 - val_accuracy: 0.8383
Epoch 2/5
100/100 [=====] - 189s 2s/step - loss: 0.7049 - accuracy: 0.7820 - val_loss: 0.2377 - val_accuracy: 0.9341
Epoch 3/5
100/100 [=====] - 175s 2s/step - loss: 0.4369 - accuracy: 0.8675 - val_loss: 0.1612 - val_accuracy: 0.9521
Epoch 4/5
100/100 [=====] - 182s 2s/step - loss: 0.2701 - accuracy: 0.9226 - val_loss: 0.1267 - val_accuracy: 0.9551
Epoch 5/5
100/100 [=====] - 184s 2s/step - loss: 0.1790 - accuracy: 0.9417 - val_loss: 0.1094 - val_accuracy: 0.9521

```

Let's check our trained model on the test images.

```

# Predict the label of the test_images
pred = model.predict(test_images)
pred = np.argmax(pred,axis=1)
# Map the Label
labels = (train_images.class_indices)
labels = dict((v,k) for k,v in labels.items())
pred1 = [labels[k] for k in pred]
pred1

'bell pepper',
'carrot',
'cauliflower',
'paprika',
'mango',
'jalepeno',
'pear',
'beetroot',
'raddish',
'sweetpotato',

```

Now we need to feed some random images from google or from the validation images. Let's create a separate function for it,

```
def output(location):
    img=load_img(location,target_size=(224,224,3))
    img=img_to_array(img)
    img=img/255
    img=np.expand_dims(img,[0])
    answer=model.predict(img)
    y_class = answer.argmax(axis=-1)
    y = " ".join(str(x) for x in y_class)
    y = int(y)
    res = labels[y]
    return res
```

```
img = output('../input/fruit-and-vegetable-image-recognition/test/cabbage/Image_1.jpg')
img
'cabbage'
```

Let's save the model so we can use it in our application.

```
model.save('FV.h5')
```

Now we need to create a web app using Streamlit.

Let's create an application code. It contains the following things.

- GUI for app
- Model processing code
- Image saving and prediction
- Fetch the calories from google for a particular class.

Now type the following command to run the code in CMD.

streamlit run Streamlit_App.py

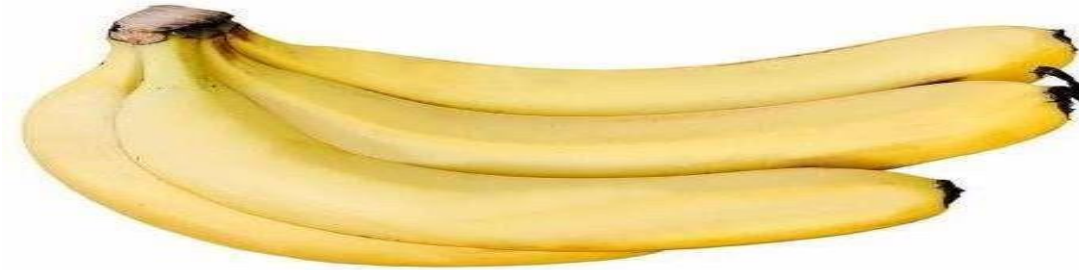
Go to the localhost, and upload the image into the app.

Display Output



Result

In this process 1st we run the main code then the GUI will display. In this we see input image and other options. Then we select the input image. Then the output will be displayed



After this there are two more predictions through mobileNet one of them is the category of the food item and second is type of food and third is most important result is the calorie of that food item.

Image Parameters

In this process 1st we take the input image of food. After selection of input image a command window will be open. This window shows the parameters of food. Then we follow this process for different- different food image and the resultant parameters are given in the table which is given below.

Table 1. MobileNet Body Architecture

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5×	Conv dw / s1	$3 \times 3 \times 512$ dw
	Conv / s1	$1 \times 1 \times 512 \times 512$
	Conv dw / s2	$3 \times 3 \times 512$ dw
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

Displayed Output

The displayed is based on Three categories:

- Taking image as an input
- Prediction of Image Category
- Prediction of Type of Image Category
- Prediction of Calorie of that Food Image

Create Web-app file into local machine. Use a saved model to recognize the image.

In this we are going to see how our web-application is working. We have divided our modules so our task is going to be easy. Our frontend-backend will be handled by the Streamlit. As a normal user, user will visit our application by URL. There will be upload button so user can upload the image. After the uploading the Image our system will do the task automatically. User, will upload the Image. That image will be stored into the local system. Now pillow will resize the image according to our model shape, it will convert into vector. Now this vector will be passed to our model, our model will classify the class of category. We will get the ID of category, now we need to map the labels according to the ID. Now our system will do web-scrap the calories for predicted object. Our application will display the Result and Calories into our application.

CHAPTER -05

5.1

CONCLUSION

The major aim of this paper is to demonstrate the better idea to detect the images of the foods and vegetables with a very efficient and light weighted algorithm. There are very large classifications of food that we take in our body. So this paper elaborates with a system that will help everyone to compute the calorie of different types of foods and vegetables with a very light weighted algorithm. This proposed system always gives very high accuracy and good performance. The image parameters work on many other factors for the recognition of food. For this system, it is very important to use a very large dataset as well. In the current scenario, there are many other algorithms who are currently working on Calorie estimation but those algorithms are very high rated and require a high weighted system as well. But the proposed system is totally dependent on MobileNet Algorithms which is one of the major algorithms of Cnn for detecting the images. One of the major benefits of this algorithm is that it is very light weighted and also provides high accuracy as compared to the previous methods.

5.2 FUTURE WORKS

Food recognition & dietary intake estimation using computer vision is an emerging field of computer engineering. Our system has demonstrated identification or classification of food from food image using image processing & artificial neural network. Our system has demonstrated decent accuracy in recognition of food. As automated food classification is an rapidly emerging field, the techniques & systems have to adapt to the pace of development & improvement & add-ons to the system are sought. One of the most sought improvements is the addition of automatic calorie estimation depending upon the food type. Other improvement can be advent of a total dietary management system based upon proposed technique which can aid in selection of food types, nutrient cycles & can comment on food selection according to physio medical Requirements.

CHALLENGES

- Recognizing the food item with the help of single picture
- Similar type of images for example roti and dosa, both are in same shape which we find difficult to recognize
- Dataset becomes much larger when it comes on food images, so currently we take a finite dataset for training

BENEFITS

- Precise and accurate recognition of food
- Rapid estimation of calorie helps users to monitor their nutritional intake
- Can keep track of dietary information

REFERENCES

- [1]: Kaylen J. Pfisterer, Robert Amelard, Audrey G. Chung, Braeden Syrnyk, Alexander, MacLean, Heather H. Keller & Alexander Wong , “Automated food intake tracking requires depth refined semantic segmentation to rectify visual volume discordance in long term care homes” published in Science Report Issue 1 Volume 2, 2022 PP 1 -13, | <https://doi.org/10.1038/s41598-021-03972-8> .
- [2]: V. H. Reddy, S. Kumari, V. Muralidharan, K. Gigoo, and B. S. Thakare, “Food Recognition and Calorie Measurement using Image Processing and Convolutional Neural Network,” published in 2020 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT), 2019, pp. 109-115. | DOI:10.1109/I2MTC.2016.7520547 | Corpus ID: 26625125
- [3]: Yogesh, Ashwani Kumar Dubey, Rajeev Ratan ,” Development of Feature Based Classification of Fruit using Deep Learning” published in International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-12, October, 2019 | Retrieval Number L28041081219 | DOI: 10.35940/ijitee.L2804.1081219

- [4]: Sharmeen M. Saleem , Subhi R. M. Zeebaree and Maiwan B. Abdulrazzaq
 ,” Real-life Dynamic Facial Expression Recognition: A Review” published in Journal of Physics:
 26-27 MAY, 2021 Conference Series, Volume 1963, 2nd International Conference on Physics
 and Applied Sciences (ICPAS 2021), College of Education, Mustansiriyah University, Baghdad,
 Iraq, Citation Sharmeen M. Saleem et al 2021 J. Phys.: Conf. Ser. 1963 012010
- [5]:Jun Zhou 1 ; Dane Bell 2 ; Sabrina Nusrat 3 ; Melanie Hingle 4 ; Mihai Surdeanu 3 ;
 Stephen Kobourov 3 ,” Calorie Estimation From Pictures of Food: Crowdsourcing Study “
 published in 2019, at JMIR Publications Advancing digital health and Open Science Published on
 5.11.2020 in Vol 7, No 2 (2020): Jul-Dec | Interact J Med Res 2018;7(2):e17 |
 doi:10.2196/ijmr.9359
- [6] L. M. r. Azizah, S. F. Umayah, S. Riyadi, C. Damarjati, and N. A. Utama, "Deep learning
 implementation using convolutional neural network in mangosteen surface defect detection,"Date
 of Conference: 24-26 Nov. 2019 Date Added to IEEE Xplore: 08 February 2019 Published in:
 2019 7th IEEE International Conference on Control System, Computing and Engineering
 (ICCSCE) , INSPEC Accession Number: 17577198 DOI: 10.1109/ICCSCE.2017.8284412
 Publisher: IEEE Conference Location: Penang, Malaysia
- [7]: V Balaji Kasyap; N. Jayapandian Food Calorie Estimation using Convolutional Neural
 Network Date of Conference: 13-14 May 2021 Date Added to IEEE Xplore: 15 June 2021
 published in 2021 3rd International Conference on Signal Processing and Communication
 (ICPSC) INSPEC Accession Number: 20843253 DOI: 10.1109/ICSPC51351.2021.9451812
 Publisher: IEEE Conference Location: Coimbatore, India
- [8]: Pramod B. Deshmukh; Vishakha A. Metre; Rahul Y. Pawar Calorimeter: Food Calorie
 Estimation using Machine Learning Date of Conference: 5-7 March 2021 Date Added to IEEE
 Xplore: 09 April 2021Published in: 2021 International Conference on Emerging Smart
 Computing and Informatics (ESCI) INSPEC Accession Number: 20553623 DOI:
 10.1109/ESCI50559.2021.9397023 Publisher: IEEE Conference Location: Pune, India

- [9]: Ma, P., Pong Lau, C., Yu, N., Li, A., Liu, P., Wang, Q., Sheng, J., Image-based Nutrient Estimation for Chinese Dishes Using Deep Learning, Food Research International (2021), PII: S0963-9969(21)00336-7 Reference: FRIN 110437 To appear in: Food Research International Received Date: 9 December 2020 Revised Date: 12 April 2021 Accepted Date: 21 May 2021 DOI: <https://doi.org/10.1016/j.foodres.2021.110437>
- [10]: Nareen O. M. Salim¹, Subhi R.M. Zeebaree¹, Mohammed A. M. Sadeeq¹, A. H. Radie², Hanan M. Shukur³ and Zryan Najat Rashid⁴ Study for Food Recognition System Using Deep Learning published in Journal of Physics:26-27 MAY, 2021 Citation Conference Series, Volume 1963, 2nd International Conference on Physics and Applied Sciences (ICPAS 2021), College of Education, Mustansiriyah University, Baghdad, Iraq, Nareen O. M. Salim et al 2021 J. Phys.: Conf. Ser. 1963 012014
- [11]: Parth Poply; J. Angel Arul Jothi Refined Image Segmentation for Calorie Estimation of Multiple-dish food items Date of Conference: 19-20 Feb. 2021 Date Added to IEEE Xplore: 12 April 2021 Published in: 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS) ISBN Information: INSPEC Accession Number: 20633673 DOI: 10.1109/ICCCIS51004.2021.9397169 Publisher: IEEE Conference Location: Greater Noida, India
- [12]: Rafhan Amnani Rahmat; Suhaili Beeran Kutty Malaysian Food Recognition using Alexnet CNN and Transfer Learning Date of Conference: 3-4 April 2021 Date Added to IEEE Xplore: 26 May 2021 Published in: 2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE) ISBN Information: INSPEC Accession Number: 20763891 DOI: 10.1109/ISCAIE51753.2021.9431833 Publisher: IEEE Conference Location: Penang, Malaysia
- [13]: Landu Jiang; Bojia Qiu; Xue Liu; Chenxi Huang; Kunhui Lin DeepFood: Food Image Analysis and Dietary Assessment via Deep Model Published in: IEEE Access (Volume: 8) Page(s): 47477 – 47489 Date of Publication: 13 February 2020 Electronic ISSN: 2169-3536 INSPEC Accession Number: 19463933 DOI: 10.1109/ACCESS.2020.2973625 Publisher: IEEE

- [14]: Mohammed A. Subhi; Sawal Hamid Md. Ali; Ahmad G. Ismail; Masuri Othman Food volume estimation based on stereo image analysis Published in: IEEE Instrumentation & Measurement Magazine (Volume: 21, Issue: 6, December 2018) Page(s): 36 – 43 Date of Publication: 13 December 2018 ISSN Information: INSPEC Accession Number: 18344296 DOI: 10.1109/MIM.2018.8573592 Publisher: IEEE
- [15]: Z. Guo, Y. Lu, H. Zhou, Z. Li, Y. Fan, and G. Yu, "Anthropometric-based clustering of pinnae and its application in personalizing HRTFs," International Journal of Industrial Ergonomics, vol. 81, p. 103076, 202 July 2021 Journal of Physics Conference Series 1963(1):012014 DOI:10.1088/1742-6596/1963/1/012014
- [16]: D. A. Zebari, H. Haron, S. R. Zeebaree, and D. Q. Zeebaree, "Enhance the Mammogram Images for Both Segmentation and Feature Extraction Using Wavelet Transform," in 2019 International Conference on Advanced Science and Engineering (ICOASE), 2019, pp. 100- 105 2nd International Conference on Physics and Applied Sciences (ICPAS 2021) Journal of Physics: Conference Series 1963 (2021) 012014 IOP Publishing doi:10.1088/1742-6596/1963/1/012014
- [17]: L. M. r. Azizah, S. F. Umayah, S. Riyadi, C. Damarjati, and N. A. Utama, "Food Estimation Using Deep learning," Published in: 2019 7th IEEE International Conference on Control System, Computing and Engineering (ICCSCE) ,Date of Conference: 24-26 Nov. 2019 Date Added to IEEE Xplore: 08 February 2019 INSPEC Accession Number: 17577198 DOI: 10.1109/ICCSCE.2017.8284412 Publisher: IEEE Conference Location: Penang, Malaysia
- [18]: V Balaji Kasyap; N. Jayapandian Food Prediction Using Machine Learning Algorithms published in 2021 3rd International Conference on Signal Processing and Communication (ICPSC) Date of Conference: 13-14 May 2021 Date Added to IEEE Xplore: 15 June 2021INSPEC Accession Number: 20843253 DOI: 10.1109/ICSPC51351.2021.9451812 Publisher: IEEE Conference Location: Coimbatore, India
- [19]: Ma, P., Pong Lau, C., Yu, N., Li, A., Liu, P., Wang, Q., Sheng, J., Image-based Food Estimation for Chinese Dishes Using Deep Learning, Food Research International (2021),

PII: S0963-9969(21)00336-7 Reference: FRIN 110437 To appear in: Food Research

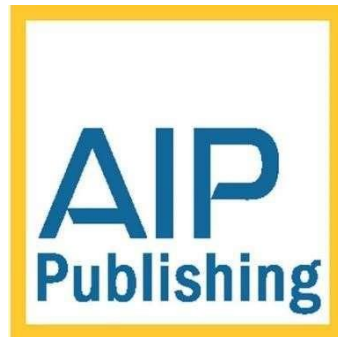
International Received Date: 9 December 2020 Revised Date: 12 April 2021 Accepted Date:

21 May 2021 DOI: <https://doi.org/10.1016/j.foodres.2021.11043>

highlight your program/ product/ institution in front of eminent from industry and academia. There are various levels of sponsorship for the Conference and recognition will be given to each sponsor at the social and business events, in pre-conference marketing materials, conference website and related activities. For more information please click the sponsorship plan.

Publication Partner

<https://aip.scitation.org/journal/apc>



❖ Indexed in leading databases — Web of Science & Scopus

ACCEPTANCE MAIL



Fee Payment and Receipt



To Bhupendra Singh Chauhan

₹8,000



Completed • February 27, 2022 at 3:05 PM



Kotak Mahindra Bank XXXXXXXXX2344



UPI transaction ID
205809243042

To
.... 3956

From: PRATYUSH SRIVASTAVA (Kotak Mahindra Bank)
pratyushsrivastava500-1@okicici

Google transaction ID
CICAgODI_NLzDA

POWERED BY 

Conference Certificates



www.ijari.org

Publication Partners



International Conference of Advance Research and Innovation ICARI-2022

CERTIFICATE OF PARTICIPATION

This is to certify that Dr./Mr./Ms. **Dr. Prashant Johri** presented/attended/volunteered paper entitled "**Food Calorie Prediction Using Mobilenet**" for "**International Conference of Advance Research and Innovation (ICARI-2022)**" organized by International Journal of Advance Research and Innovation (www.ijari.org), held online on 5th March 2022.

We express our heartfelt gratitude to him/her for sparing valuable time and enabling us in successfully organizing ICARI-2022.



Ramakant Rana
Co-Convener



Dr. Bhupendra Singh Chauhan
Conference Chair



www.ijari.org

Publication Partners



International Conference of Advance Research and Innovation ICARI-2022

CERTIFICATE OF PARTICIPATION

This is to certify that Dr./Mr./Ms. **Pratyush Srivastava** presented/attended/volunteered paper entitled "**Food Calorie Prediction Using Mobilenet**" for "**International Conference of Advance Research and Innovation (ICARI-2022)**" organized by International Journal of Advance Research and Innovation (www.ijari.org), held online on 5th March 2022.

We express our heartfelt gratitude to him/her for sparing valuable time and enabling us in successfully organizing ICARI-2022.



Ramakant Rana
Co-Convener



Dr. Bhupendra Singh Chauhan
Conference Chair

AIP



www.ijari.org

Publication Partners



International Conference of Advance Research and Innovation ICARI-2022

CERTIFICATE OF PARTICIPATION

This is to certify that Dr./Mr./Ms. **Abhishek Tiwari** presented/attended/volunteered paper entitled "**Food Calorie Prediction Using Mobilenet**" for "**International Conference of Advance Research and Innovation (ICARI-2022)**" organized by International Journal of Advance Research and Innovation (www.ijari.org), held online on 5th March 2022.

We express our heartfelt gratitude to him/her for sparing valuable time and enabling us in successfully organizing ICARI-2022.

A handwritten signature in black ink, appearing to read "Rana...".

Ramakant Rana
Co-Convener

A handwritten signature in blue ink, appearing to read "Bhupendra Singh Chauhan".

Dr. Bhupendra Singh Chauhan
Conference Chair