**FOOD DETECTION WITH IMAGE PROCESSING USING CONVOLUTIONAL NEURAL NETWORK (CNN) METHOD**

**Abstract**

Currently, the payment process at restaurants is still manual and inefficient because it uses a cash register. A cashier will check what food is ordered, then count it with the cash register. This is not efficient. So food detection devices and automatic food price estimates have the answer to these deficiencies. Food detection aims to facilitate payment at restaurants, and automatic food price estimation using the Convolutional Neural Network (CNN) classification method. **The detection accuracy of 6 types of food using the CNN method was obtained 100% with 80% data partition training data and 20% test data with epoch 9000 and learning rate 0.0002, with a detection time of fewer than 10 seconds.**

The food Detection System is the representation of advanced stages by explaining what food is in one frame of a picture taken.

This is implied in food analysis in restaurants and auto-billing.

The CNN used by Aguillar et al. had 90% accuracy thus, very efficient and has higher accuracy than the traditional vector submachine-based method with handcrafted features that the convolution kernels show that color dominates the feature extraction process.

2. Dataset:

The dataset was gathered according to the food that is often bought at the canteen.

80% training data; 20% test data;

epoch=9000; Epoch is the one entire passing of training data through the algorithm. or (It's a hyperparameter that determines the process of training the machine learning model)

learning rate =0.0002

**Limitation:**

The data is not large enough to predict with so much accuracy.

**IMAGE PROCESSING**

Before the images are managed to create models, pre-processing is carried out to make the data set lighter for processing by the computer, rotation, color changes, and pixel size reduction are carried out in this process.

1. Sampling is the process of determining the color of specific pixels in an image. (pixel size). This is the process of extracting subsets of information from an input weight. This is done by reducing computational cost of processing large images. Examples includes the max pooling and average pooling. **Max Pooling** involves partitioning the input feature map into non-overlapping rectangular regions, and then selecting the maximum value from each region. **Average Pooling** instead of selecting the maximum value from each region, it computes the average value. However, average pooling is seldomly used.

ii. Quantization associating average colors to certain color levels)

iii. Noise: an image or pixel that interferes with image quality.

**CNN**

Convolutional Neural Network (CNN) is a method in deep learning that is used for image classification, semantic segmentation, object detection, and feature extraction.

CNN is included in the type of deep neural network because of the high network depth and is widely applied to image data.

**REASON WHY THEY CHOOSE CNN**

Food references can be found by way of food classification. Image classification is challenging because the food image dataset is not linear; for example, there is one food with another, which is not the same, and there is also one food with another food that has similarities in food types and shapes. This CNN will be efficient to help extract the features and semantically segment the images.

**RELATED WORKS**

1. **Estimating Food Calories for Multiple-Dish Food Photos.**

Food has a calorie; in a meal, the consumer usually does not know how many calories are in a food that is served. Food detection is needed. In a food photograph generally includes several types of food served, and in one picture can be detected what food is served.

1. **Deep Structured Convolutional Neural Network for Tomato Diseases Detection.**

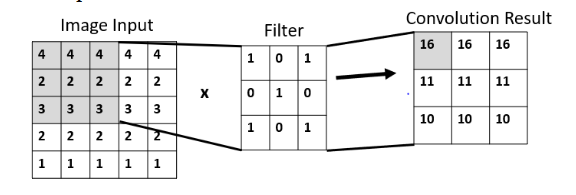
Outbreaks of plant diseases can cause threats to food security. Technology is needed for machine learning and for recognizing objects. CNN is one technique or method for object detection and object identification in depth. CNN also has architectures such as AlexNet and VGGNet. These architectures help to capture information on various lengths and as well capture correlation between pixels. These architectures are useful to modify the number of filters and depth or width to the network. CNN accuracy without adding those architecture is 89% however, adding VGGNet rises to 95.24%

1. **Image Recognition with Deep Learning**

Foods have many characteristics to be classified, and the classification method can use existing methods in computer vision. CNN is a neural network class that is very effective for the task of image classification and object detection. Choosing foods is essential for specific purposes, such as a diet rating system, to avoid obesity, diabetes, and others. This could be possible by way of classification.

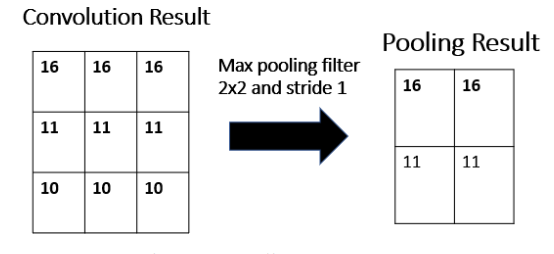
**CONVOLUTION LAYER**

It finds the texture/pattern in the image by conducting the convolution processing between the filter and the image. The convolution layer located in the inception architecture, used a 5x5 dimension image and a 3x3 filter, and a 1 shifter filter or pixel stride. The sliding window (filter) will be passed over the image input.



**POOLING LAYER**

This layer is responsible for partitioning/dividing several parts in a square image. The MAX part of the convolution layer is taken. Thus, by using the MAX-POOLING filter (2x2) and a stride 1 (thus shifting the pixel by 1). The max-pooling technique applies a max filter to nonoverlapping sub-regions of the input features. This is used to reduce the dimensionality of the features. Drop-Out Layer: I think this pooling was used to improve over-fit on the neural network.

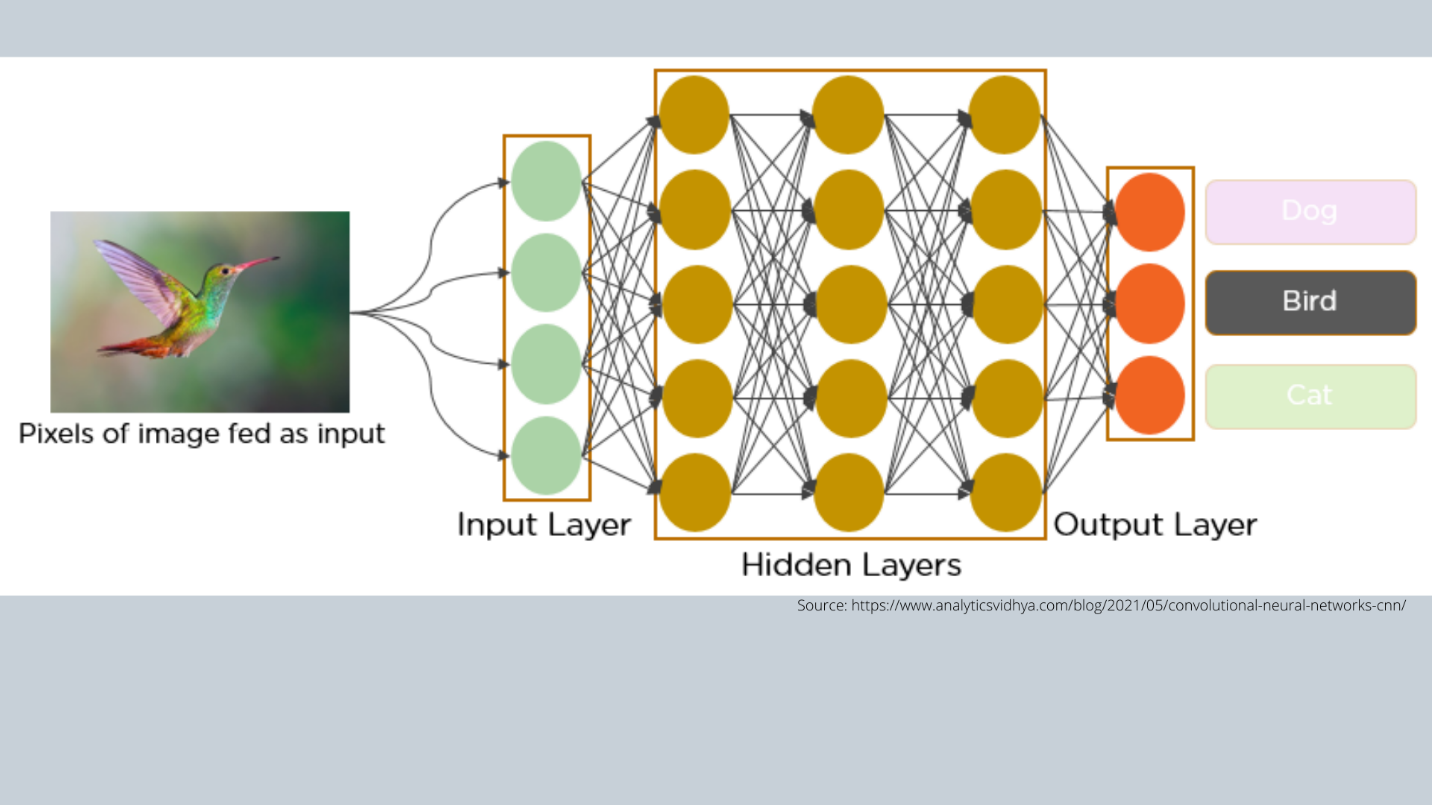


The results of the pooling layer obtained the most significant values. The results of this layer will proceed to the next process, the fully connected.

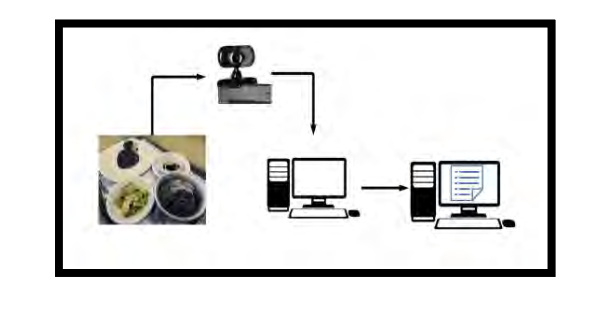
**FULLY CONNECTED LAYER**

**This stores the whole layers’ results by connecting other neurons to one neuron and storing its shape.** This is a dense connection of neurons, and connection between every two neurons.

After the images has passed through the convolutional layer, we then pass it to the fully connected layer.

Not that, in this the images are flatten into a 1-dimensional array. Afterwards the flatten array will be used as input to the fully connected layer. The number of neurons in the last fully connected layer has the same output as the output image. The connection between the neurons are called **weight**, which are trainable parameters that the model has to learn. This is applied after the convolution and max pooling layers. This helps to classify output. And also associate features to a particular label.

**TRIANING THE MODEL**



This shows the process of food detection. There should be a dataset to be recognized. The dataset will be in the process of training and testing.

WEBCAM:

It was useful to use a webcam since it could easily be incorporated into other mobile devices. Also; it provides the right image and a tripod.

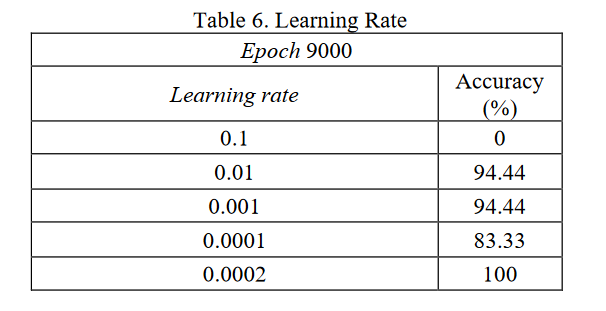
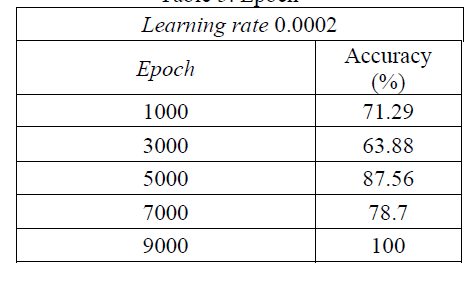
**DATA PARTITION TESTING**

This is a function used to train the system to recognize or detect the image.

420 images: 80% training images and 20% testing images. 9000 epoch and 0.0002 learning rate.

**Epoch sample** trained to recognize the best accuracy value.

**Learning Rate**: the learning rate is a tuning parameter or hyperparameter in an optimization algorithm that determines or controls the step size at each iteration while moving toward a minimum loss function.



**Epoch and Learning Rate** For having the best accuracy of food detection. CNN must have to test the epoch and learning rate for the best system to recognize the food image. Epoch is compiling the dataset all through one time forward and backward process through all the neural network nodes 1 time. The learning rate serves to get the highest accuracy. It controls how quickly the model adapts to the problem and how quickly it converges to the optimal solution.

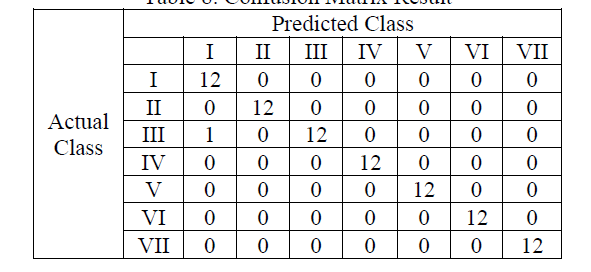
CONFUSING MATRIX

This is used to measure/evaluate the performance of a classification algorithm. Thus, it summarizes the prediction made by the model and compares them to the actual results from the inputs. The matrix represents the number of times a sample was categorized correctly and incorrectly. It has four basic elements. Thus:

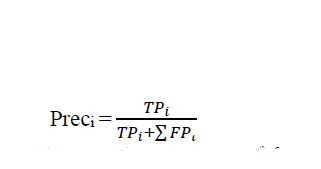
1. TP: number of samples that were correctly predicted as positive.
2. TN: number of samples that were correctly predicted as negative.
3. FP: number of samples that were incorrectly predicted as positive
4. FN: number of samples that were incorrectly predicted as negative

NB: The True, and False represent the predicted values of the model. And the Positive and Negative represents the actual values.

Based on these values we would be able to calculate the precision, accuracy, F1-Score, and recall.







**augment the training set, and fine-tune the hyperparameters.**

Data augmentation involves applying random transformations to the images, such as flipping, rotating, and zooming, to create new variations of the images.

**All of this hyperparameters fine-tuned produced a 95% accuracy.**

ADDITIONS

IMAGE PREPROCESSING

**Again this study uses zero-phase component analysis (ZCA) whitening, to reduce image matrix redundancy, and automatically rotates the image randomly in the training process from 0 to 180 degrees. To handle the difference in image position a shifted version of the training data will be, where each image will be shifted either horizontally or vertically.**

**NOTE: Inception version three (V3)**

**NOTES FROM CNN TUTORIAL**

This can be an overview of the presentation.

The concept behind CNN is using filters which are sliding windows on the images responsible for detecting the features or patterns in the image. So, for the filter in CNN, the first filter layer might detect patterns the next layer the nose and eyes, and the subsequent layer might detect the entire face. After it can be associated with a particular label. (Human, or a dog, cat, etc)

The filters are superimposed on the original image matrix.

**Padding**

Without padding the image, the edges do not get match attention whiles the middles get so much attention. Also, when the filter is applied, the size of the image (matrix) is reduced drastically.

For instance, if an original image has a 6 x 6 matrix without padding and a filter of 3 x 3 is applied the results will be a 4x4 image, thus a reduction in the image size. However, if the original image (6 x6) is padded, it size will be 8 x 8, and a filter of 3 x 3 applied will give a 6 x 6 matrix therefore there is no or just some little detail of the image will be missing. Once more, the first image is exposed to a lot more times compared to the non-padded image.

“VALID” and “SAME” Convolution

Valid Convolution: No padding

Same convolution.

**TensorFlow implementation of “VALID/ SAME” convolution**

Tf.nn.conc2d(x, W1, strides =[1,1,1,1], padding=”VALID/SAME”)

**Strides (Strided Convolution)**

This is how many pixels to move the filter across the various pixels of the original image. If Stride = 1, means move the filter by one pixel, stride = 2, means move the filter by two pixels etc. Incase there are no enough pixels at the tail end, those once are left out or discarded.

**Max Pooling Layer**

This reduces the size or dimension of the image. For max pooling usually, the Stride is taken as the side length of the filter. So if the filter is 2 x 2 then the Strid could be 2. Note that the resulting image will be a 2 x 2 image. This is how it works, the filter window is placed on top of the top left matrix of the original image, the max value from the original image is chosen and recorded for the output pixel. This is repeated until all the image pixels are exhausted. NB: we will stride the filter by the stride about which in our case is 2.

**Why Max Pooling**

1. Reduces the image size and reduces computation cost. The model can be trained faster too.
2. It also sharpens or enhances the features of the image. Since the max pixel is taken that tends to have the image feature after all are combined together.

Max pooling is often applied after the convolutional layer. This improves the performance of the model and it is very powerful as well. No parameters are involved when performing the max pooling layer.

NB: Average Pooling: The average of the values are taken in one filter and recorded as the output.

**Full Connected Layer**

This is a dense connection of neurons and a connection between every two neurons.

After the images have passed through the convolutional layer we then pass it to the fully connected layer.

Not that, in this the images are flattened into a 1-dimensional array. Afterward, the flattened array will be used as input to the fully connected layer. The number of neurons in the last fully connected layer has the same output as the output image. The connection between the neurons is called **weight**, which are trainable parameters that the model has to learn. This is applied after the convolution and max pooling layers. This helps to classify output. And also associate features with a particular label.

All the above are used to create a **forward propagation** in the convolutional neural network.

**SoftMax activation function.** Multiclass classification. The categorical cross-entropy cost function can be used.

**Sigmoid activation:**  Used when dealing with a single classification. (Binary classification). The Binary cross entropy ***Cost function*** can be used. In minimizing the cost function to increase the accuracy of the model we use Backpropagation such as the stochastic Gradient Decent, or Adam, Momentum

No linear activation function (ReLU)