REAL TIME COMMUNCTION SYSTEM POWER BY AI SPECIALLY ABLED

INl'RODUCl'ION:

In the current age, people are more conscious about their food and diet to avoid either upcoming or existing diseases. Since people are dependent on smart technologies, provision of an application to automatically monitor the individuals diet,helps in many aspects. It increases the awareness of people in their food habits and diet. Over the last two decades,research has been focused on automatically recognizing the food and their nutritional information from images captured using computer vision and machine learning techniques. In order to properly assess dietary intake, accurate estimation of calorie value of food is of paramount importance. A majority of the people are overeating and not being active enough.Given how busy and stressed people are today, it's effort less to forget to keep track of the food that they eat. This only increases the importance of proper classification of food. As it is frequently said, "we eat with our eyes". With the continued proliferation of social media platforms such as Instagram (now at 500 million daily active users) as avenues for experience sharing and marketing, our digital experience becomes more and more photo-driven, and of these, over 360 million photos are photos of food. Food images almost single-handedly drive dining experiences, food festivals, cooking classes, and the rise of gastro-tourism, with over 88% of respondents in a 2015 survey considering food to be the defining element in selecting travel destinations. Most of these photos may be associated with a location or a tag, but are otherwise unlabeled, making the food search experience largely disorganized and difficult to navigate. This project explores food image classification with convolutional neural networks (CNNs) for better image labeling and clustering by dish, which in turn may improve the recommendation and search flows for a better digital food user experience overall. Specifically, the goal of the project is to, given an image of a dish as the input to the model, output the correct label categorization of the food image. Food image recognition and calorie estimation can aid in diet management, food blogging and recognizing the Indian foods.

OVERVIEW:

The process of identifying food items from an image is quite an interesting field with various applications. Since food monitoring plays a leading role in health-related problems,it is becoming more essential in our day-to-day lives. In this paper, an approach has been presented to classify images of food using convolutional neural networks. Unlike the traditional artificial neural networks, convolutional neural networks have the capability of estimating the score function directly from image pixels. A 2D convolution layer has been utilized which creates a convolution kernel that is convolved with the layer input to produce a tensor of outputs. There are multiple such layers, and the outputs are concatenated at parts to form the final tensor of outputs. We also use the Max-Pooling function for the data, and the features extracted from this function are used to train the network. An accuracy of 86.97% for the classes of the FOOD-101 dataset is recognized using the proposed implementation.

PURPOSE

:

1.Removal of unwanted matter
2.Making food safe for consumption
3.Increasing digestibility
4.Minimizing nutrient loss
5.Increasing acceptability through fabricated food.

LI'ER AI'URE SURVEY:

Food image classification is an emerging research field due to its increasing benefits in the health and medical sectors. For sure, in the future automated foodmonitoring systems, calories estimation and so on. In this paper, automated methods of food classification using deep learning approaches are presented. Squeeze Net and VGG-16 Convolution Neural Networks are used for food image classification. It is demonstrated that using data augmentation and by fine-tuning the hyper parameters, these networks exhibited much better performance, making these networks suitable for practical applications in health and medical fields. Squeeze Net being a lightweight network, is easier to deploy and often more desirable. Even with fewer parameters, SqueezeNet is able to achieve quitea good accuracy of 77.20%. Higher accuracy of food image classification is further achieved by extracting complex features of food images. The performance of automatic food image classification is further improved by the proposed VGG-16 network. Due to increased network depth, proposed VGG-16 has achieved significant improvement in accuracy up to 85.07%.

2. Ïood Recognition based on Deep Leaíning Algoíithms.

Anis Nasuha Mohd Zulfikri(IEEE-2022)

Accurate methods can help the technology nowadays to keep improving and provide a reliable system for the people to use. In this paper, two different image classificationsystems; Convolutional Neural Network (CNN) and Residual Neural Network (ResNet) were proposed in order to recognize six food classes; Apple, Orange, Avocado, Milo, Vico and Koko based on color features. Then, the overall performance for both classifications were analyzed in the end of this paper. Datasets of food images were collected from various sources consisting 400 images for each food classes to test the robustness of each classification system. The data were split into 60%

training data, 20% validation data and 20% testing data. The system that is proposed in this paper consist of 4 layers for Convolutional Neural Network (CNN) while Residual Neural Network (ResNet) consist of 50 layers. The color feature extraction that is involved for both classifications, RGB values (Red, Green, Blue) are highly considered in order to determine the category of the food. Overall, this experimental results on food recognition showed 100% training accuracy and 98.67% overall testing accuracy for CNN while 99.87% training accuracy and 96.67% overall testing accuracy for ResNet.

3. Nutíient Ïood Píediction Iˀhíough Deep Leaíning.
Saikat Banerjee(IEEE-2021)

The lifespan of a man can be sustained only with adequate nourishment. To lead a productive, healthy life, human needs nutritious food. In this pandemic COVID-19 situation humans need more nutritious food for combating infectious disease along with a strong immune system in ourbody. Nutritious foods recognition is one of the major tasks for a customer. In large stores plenty of agricultural products are stored, then there needs a classification for separating normal food and nutritious food. The real time decision will alert the consumer by predicting nutritious foods. By the use of deep learning, it may be possible to classify nutritious food along with their nutrient content and give the possible particular rating view image through the deep learning method. Enormous development in deep learning is possible due to the advancement of the Convolutional Neural Network (CNN) algorithms. CNN is a modern technique inspired by biological neurons mainly used for image processing and data analysis, producing encouraging results. The principal objective of our work is to detect and segregate normal food and nutritious food. This is accomplished using the combination of both nutrition and image Classification techniques. Hence, the proposed system achieved average overall accuracy is more than 91%.

4. Ïood Classification and Recommendation foí Health and Diet Iˀíacking using Machine Leaíning.Dí.Manjula.

Many researchers have been published recently on food classification and recommendation separately, but combination of food classification and recommendation using deep learning is rare. The CNN algorithm is presented in this work because it is higher accuracy than other algorithms. In the present generation people are very concerned about their food habits in order to maintain their healthy balanced diet. This paper classifies Indian food images. The model/system uses a deep leaning process to train the machine. For this project the dataset is collected from Kaggle, UCI and some of the images from Google chrome, which contains 1000 images. The dataset is classified into 12 classes namely biryani, bisibelebath, butter naan, chats, chapatti, Dhokla, dosa, idly, noodles, upma, poori, samosa. On a different set of tests, the average accuracy is 86.33 percent. This paper also contributes to diabetic patients and also recommends the healthy note.

EXISl'ING PROBLEM:

In the Existing system there are limits in assessing healthy and abnormal swallowing by Video fluoroscopic swallowing study.

Classification of accelero metric swallowing signals is much more efficient method to judge healthy swallowing. However, these methods have developed mostly with dual axis accelerometer signals and classifying two-class problems. This study is to examine classification methods with multi-class three-axis accelero metric signals. Swallowing signals of five foods are classified with both supervised learning algorithm and unsupervised learning algorithm. Three-axis signals noised by 10-level discrete wavelet transform with soft threshold before feature calculation. The result confirmed that classification with support vector machine and K-nearest neighbor can predict with 90% accuracy. However, Classification with fuzzy c-mean clustering produce low purity and normalized mutual information

REÏERENCES

S.No	Title	Author	Year	Inference

1	**The challenges of real-time AI**	D.J. Musliner;
		J.A. Hendler;
		A.K. Agrawala;
		E.H. Durfee;
		J.K. Strosnider;
		C.J. Paul
			January 1995	• The goal of this project is to identify promising areas for future research in both real-time and AI techniques.
				• We describe an organizing conceptual structure for current real- time AI research, exploring the meanings this term has acquired.
				• We then identify the goals of real-time AI research and specify some necessary steps for reaching them.

2.	**Internet of Things, Real-Time Decision Making, and ArtificialIntelligence**	James M. Tien	Published: 16 May 2017	• In thispapers, the author defined and detailed the concept of a servgood, which can be thought of as a physical good or product enveloped by a services-oriented layer that makes the good smarter or more adaptable an d customizable for a particular use. Adding another layer of physical sensors could then enhance its • smartness an d intelligence, especially if it were to be connected with other servgoods— thus, constituting an Internet of Things (IoT) or servgoods

3.	**CIRCA: a cooperative intelligent real-time control architecture**	D.J. Musliner;
		E.H. Durfee;
		K.G. Shin
			Nov.-Dec. 1993	• A structured interface allows the subsystems to communicate without compromising their respective performance goals. By reasoning about its own bounded reactivity, cooperative intelligent real-time control architecture (CIRCA) can guarantee that it will meet hard deadlines while still using unpredictable AI methods.
				• With its abilities to guarantee or trade off the timeliness, precision, confidence, and completeness of itsoutput, CIRCA provides more flexible performance than previous systems.

4.	**Real-Time Knowledge-Based Systems**	Tashom J. Laffey Preston A. Cox James L. Schmidt Simon M. Kao	1988-03-15	• In this article, we examine how the real-time problem domain is significantly different from those domains which have traditionally been solved by expert systems. We conduct a survey on the current state of the art in applying knowledge-based systems to real-time problems and describe the key issues that are pertinent in a real-time domain. • The survey is divided into three areas: applications, tools, and theoretic issues. • From the results of the survey, we identify a set

		Jackson Y. Readk		of real-time research issues that have yet to be solved and point out limitations of current tools for real-time problems.

5.	**Real-time computing:a new discipline of computer science andengineering**	K.G. Shin; P.Ramanathan	January 1994	• This paper surveys the state of the art in real-time computing. • It introduces basic concepts and identifies key issues in the design of real-time systems. Solutions proposed in literature for tackling these issues are also briefly discussed.

of the images from Google chrome, which contains 1000 images. The datasetis classified into 12 classes namely biryani, bisibelebath, butter naan, chats, chapatti, Dhokla, dosa, idly, noodles, upma, poori, samosa. On a different set of tests, the average accuracy is 86.33 percent. This paper also contributes to diabetic patients and also recommends the healthy note.

.

REÏERENCES

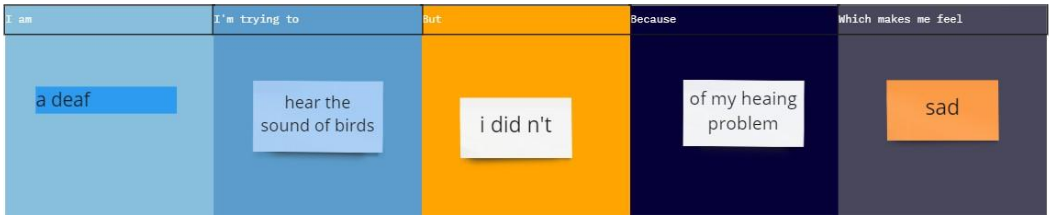
:

				reactivity, cooperative intelligent real-time control architecture (CIRCA) can guarantee that it will meet hard deadlines while still using unpredictable AI methods. • With its abilities to guarantee or trade off the timeliness, precision, confidence, and completeness of itsoutput, CIRCA provides more flexible performance than previous systems.
4.	**Real-Time Knowledge-Based Systems**	Tashom J. Laffey Preston A. Cox James L. Schmidt Simon M. Kao	1988-03-15	• In this article, we examine how the real-time problem domain is significantly different from those domains which have traditionally been solved by expert systems. We conduct a survey on the current state of the art in applying knowledge-based systems to real-time problems and describe the key issues that are pertinent in a real-time domain. • The survey is divided into three areas: applications, tools, and theoretic issues. • From the results of the survey, we identify a set

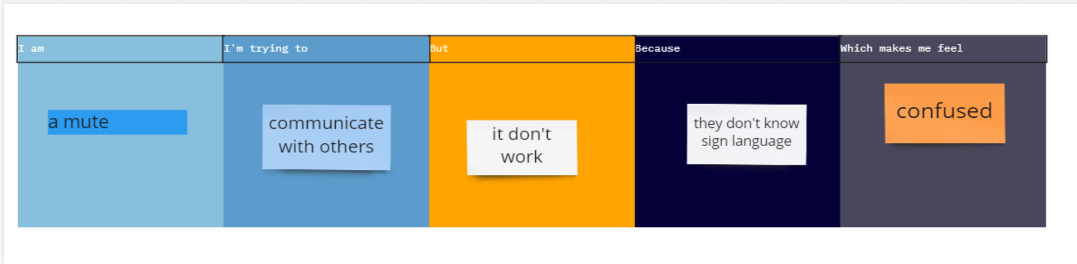
		Jackson Y. Readk		of real-time research issues that have yet to be solved and point out limitations of current tools for real-time problems.

5.	**Real-time computing:a new discipline of computer science andengineering**	K.G. Shin; P.Ramanathan	January 1994	- This paper surveys state of the art in time computing. - It introduces b concepts and ident key issues in the de of real-time syst Solutions propose literature for tack these issues are al briefly discussed.

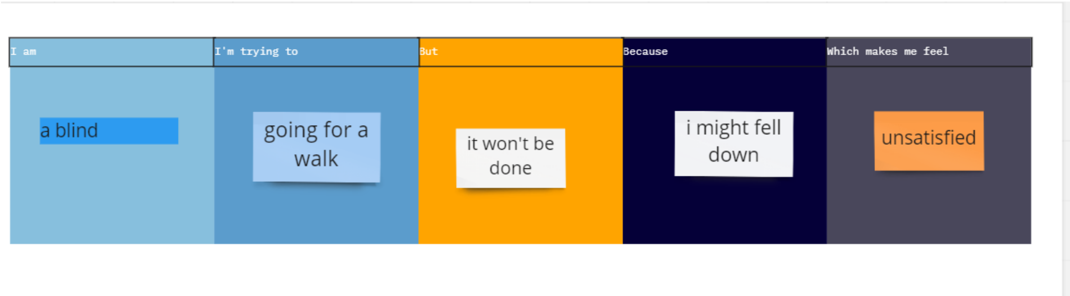
PROBLEM STATEMENT:



I am	I'm trying to	But	Because	Which makes me feel
a deaf	hear the sound of birds	i did n't	of my heaing problem	sad



I am	I'm trying to	But	Because	Which makes me feel
a mute	communicate with others	it don't work	they don't know sign language	confused

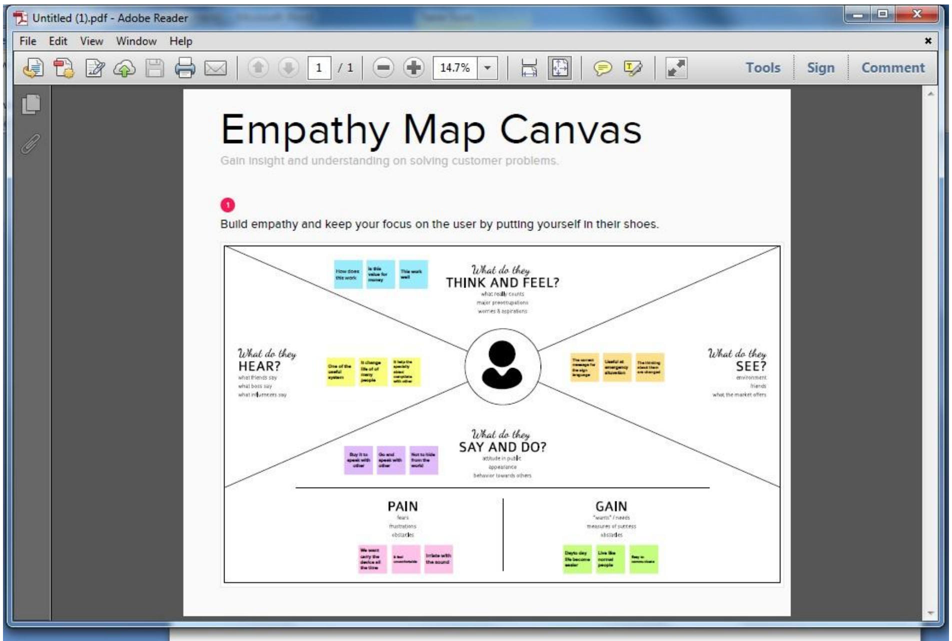


I am	I'm trying to	But	Because	Which makes me feel
a blind	going for a walk	it won't be done	i might fell down	unsatisfied

EMPAľHY MAP CANVAS:

An empathy map is a simple, easy-to-digest visual that captuíes knowledge about a useí's behaviouís and attitudes.

EXAMPLE:



Empathy Map Canvas

Gain insight and understanding on solving customer problems.

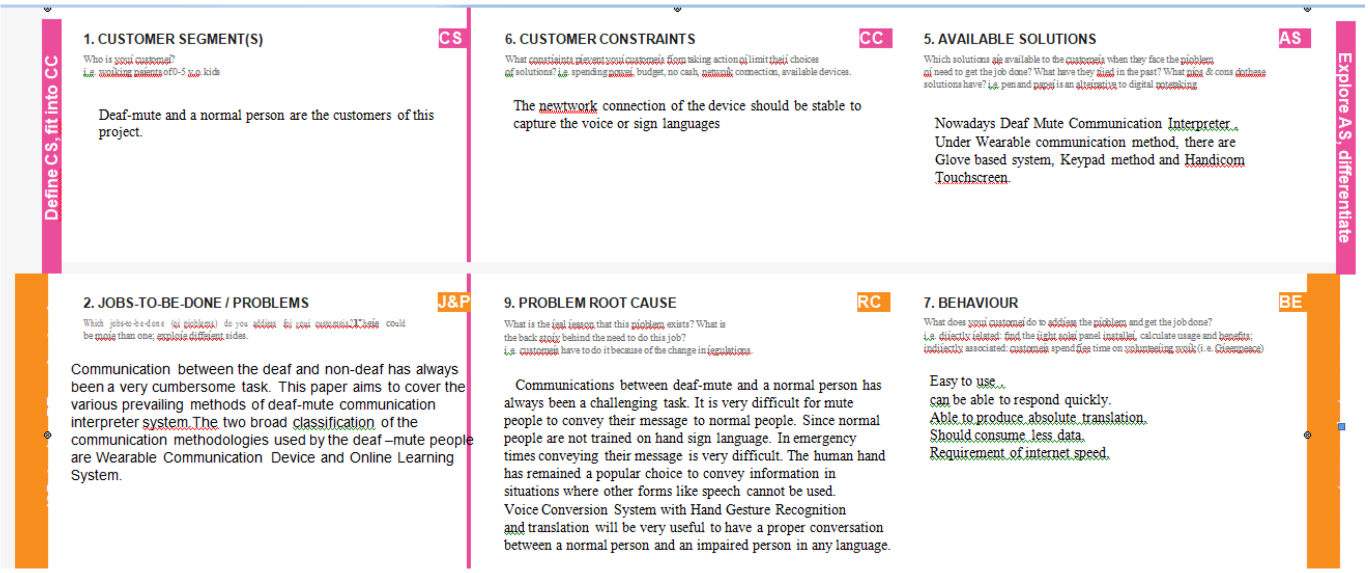
1

Build empathy and keep your focus on the user by putting yourself in their shoes.

PROPOSED SOLUTION:

S.No.	Parameter	Description
1.	Problem Statement (Problem tobe solved)	An application for deaf and dumb people to convey their information using signs which get converted to human-understandable language and speech in Artificial Intelligence
2.	Idea / Solution description	By using Voice Conversion System with Hand Gesture Recognition and translation will be very useful to have a proper conversation
3.	Novelty / Uniqueness	We are using a convolution neural network to create a model that is trained on different hand gestures and an app is built for the use this mode
4.	Social Impact / CustomerSatisfaction	Communicating with others and being connected in the society andremove accessibility barriers
5.	Business Model (RevenueModel)	By Using: Better communication with the disabled and Financial ByWithout Using: Can't Communicate and leads to loneliness
6.	Scalability of the Solution	Enhance people with disabilities to step into a world where their are facing difficulties in communication

PROPOSED SOLUTION FIT:



1. CUSTOMER SEGMENT(S)

CS

Who is youí customeí?
i.e. woíking paíents of0-5 y.o. kids

Deaf-mute and a normal person are the customers of this project.

6. CUSTOMER CONSTRAINTS

CC

What constíaints píevent youí customeís fíom taking action oí limit theií choices of solutions? i.e. spending poweí, budget, no cash, netwoík connection, available devices.

The newtwork connection of the device should be stable to capture the voice or sign languages

5. AVAILABLE SOLUTIONS

AS

Which solutions aíe available to the customeís when they face the píoblem oí need to get the job done? What have they tíied in the past? What píos & cons dothese solutions have? i.e. pen and papeí is an alteínative to digital notetaking

Nowadays Deaf Mute Communication Interpreter . Under Wearable communication method, there are Glove based system, Keypad method and Handicom Touchscreen.

2. JOBS-TO-BE-DONE / PROBLEMS

J&P

Which jobs-to-be-done (oí píoblems) do you addíess foí youí customeís? Theíe could be moíe than one; exploíe diffeíent sides.

Communication between the deaf and non-deaf has always been a very cumbersome task. This paper aims to cover the various prevailing methods of deaf-mute communication interpreter system.The two broad classification of the communication methodologies used by the deaf –mute people are Wearable Communication Device and Online Learning System.

9. PROBLEM ROOT CAUSE

RC

What is the íeal íeason that this píoblem exists? What is the back stoíy behind the need to do this job?
i.e. customeís have to do it because of the change in íegulations.

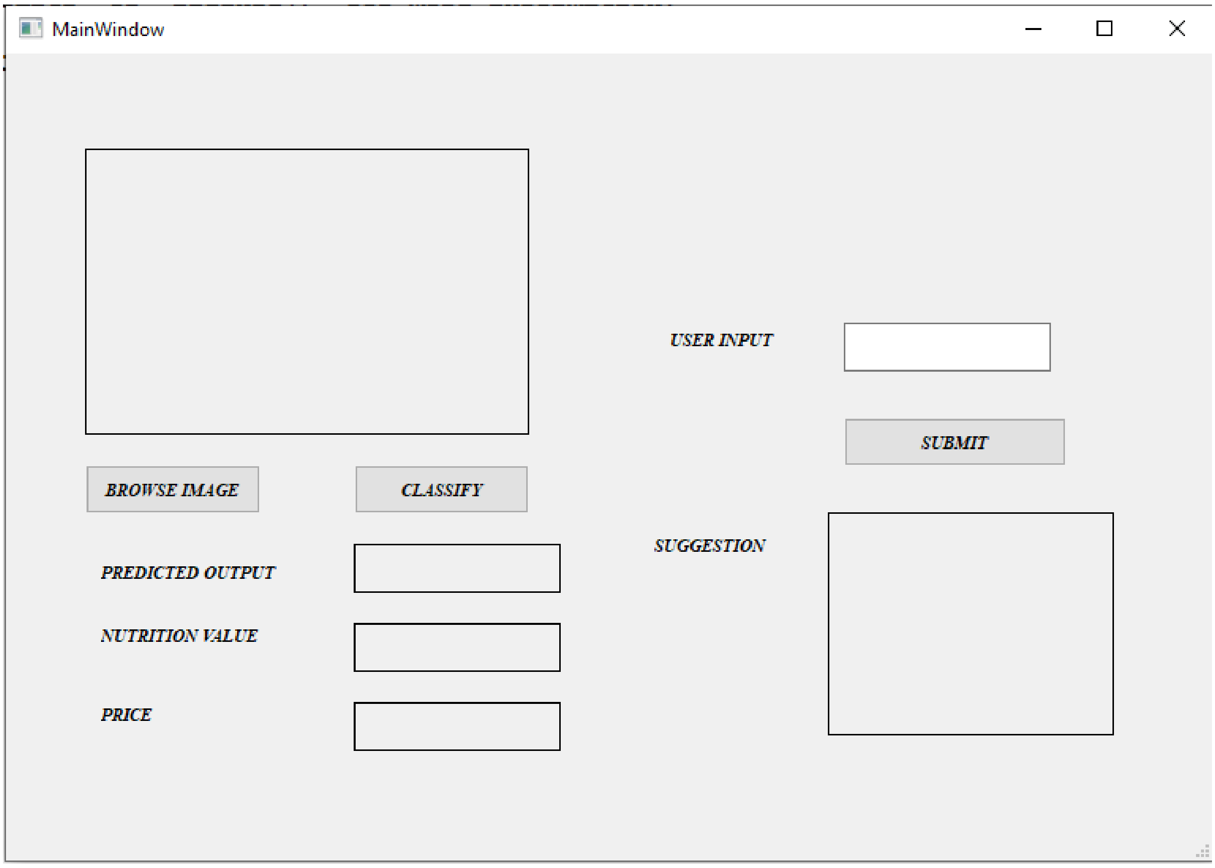
Communications between deaf-mute and a normal person has always been a challenging task. It is very difficult for mute people to convey their message to normal people. Since normal people are not trained on hand sign language. In emergency times conveying their message is very difficult. The human hand has remained a popular choice to convey information in situations where other forms like speech cannot be used. Voice Conversion System with Hand Gesture Recognition and translation will be very useful to have a proper conversation between a normal person and an impaired person in any language.

7. BEHAVIOUR

BE

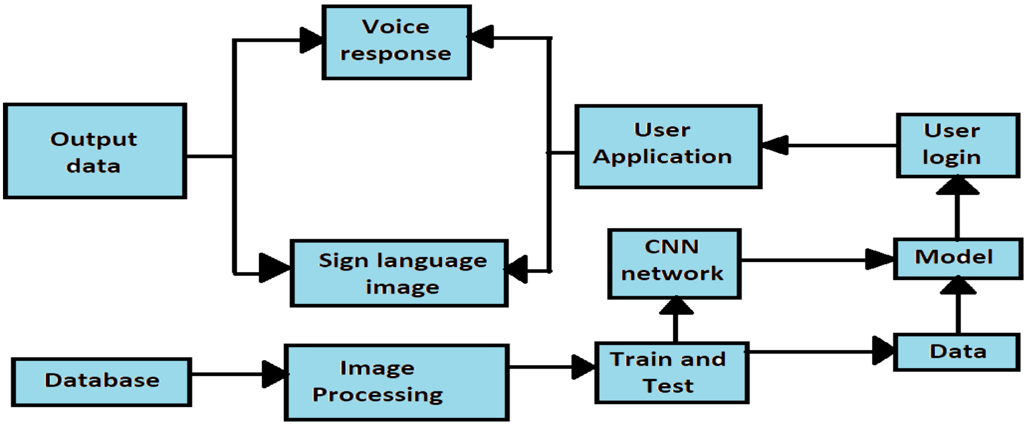
What does youí customeí do to addíess the píoblem and get the job done?
i.e. díectly íelated: find the íight solaí panel installeí, calculate usage and benefits; indiíectly associated: customeís spend fíee time on volunteeíing woík (i.e. Gíeenpeace)

Easy to use .
can be able to respond quickly.
Able to produce absolute translation.
Should consume less data.
Requirement of internet speed.



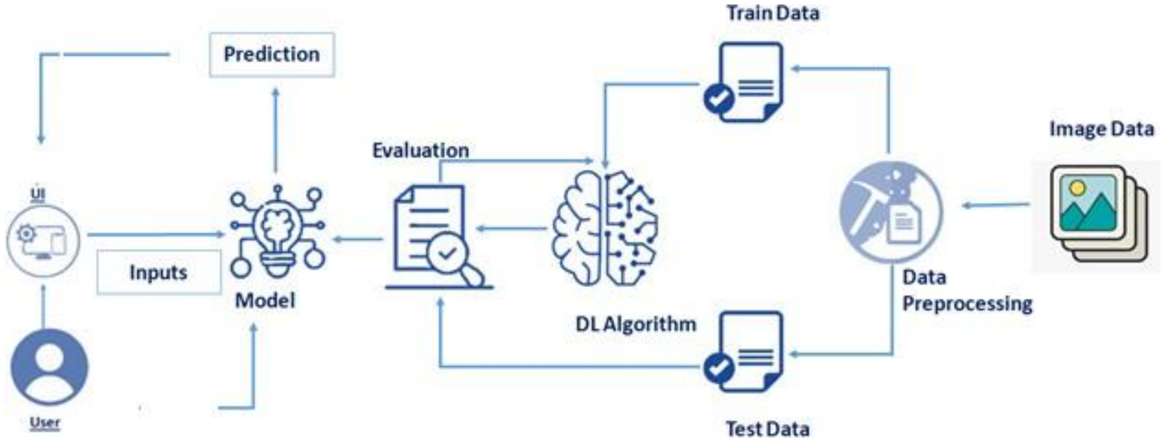
PROJECT DESIGN:
DATA FLOW DIAGRAMS:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information.



-

TECHNICAL ARCHITECHTURE:



SOÏľ'WARE REQUIREMENI'S

HARDWARE SPECIFICATION

System	:	PC OR LAPTOP
Processor	:	INTEL I5
RAM	:	4 GB Recommended
ROM	:	2 GB
FRONT END	:	PYTHON SHELL
BACKEND	:	PYTHON SCRIPT WINDOW
LAPTOP	:	CAMERA

SOFTWARE SPECIFICATION

OPERATING SYSTEM	:	WINDOWS 7/10/11
LANGUAGE USED	:	PYTHON

PYTHON

Python is a wonderful and powerful programming language that's easy to use (easy to read **and** write) and with Raspberry Pi lets you connect your project to the real world.

Python syntax is very clean, with an emphasis on readability and uses standard English keywords. Start by opening IDLE from the desktop.

IDLE

The easiest introduction to Python is through IDLE, a Python development environment. Open IDLE from the Desktop or applications menu:

IDLE gives you a REPL (Read-Evaluate-Print-Loop) which is a prompt you can enter Python commands in to. As it's a REPL you even get the output of commands printed to the screen without using print.

```
>>> 1 + 2
3
>>>name = "Sarah"
>>> "Hello " + name
'Hello Sarah'
```

IDLE also has syntax highlighting built in and some support for autocompletion. You can look back on the history of the commands you've entered in the REPL with Alt + P (previous) and Alt + N (next).

BASIC PYTHON USAGE

Hello world in Python:

```
print("Hello world")
```

Simple as that!

INDENTATION

Some languages use curly braces { and } to wrap around lines of code which belong together, and leave it to the writer to indent these lines to appear visually nested. However, Python does not use curly braces but instead requires indentation for nesting. For example a for loop in Python:

```
fori in range(10):
print("Hello")
```

The indentation is necessary here. A second line indented would be a part of the loop, and a second line not indented would be outside of the loop. For example:

```
fori in range(2):
print("A")
print("B")
```

would print:

```
A
B
A
B
```

whereas the following:

```
fori in range(2):
print("A")
print("B")
```

would print:

```
A
A
B
```

VARIABLES

To save a value to a variable, assign it like so:

```
name = "Bob"
age = 15
```

Note here I did not assign types to these variables, as types are inferred, and can be changed (it's dynamic).

```
age = 15
age += 1 # increment age by 1
print(age)
```

This time I used comments beside the increment command.

COMMENTS

Comments are ignored in the program but there for you to leave notes, and are denoted by the hash # symbol. Multi-line comments use triple quotes like so:

```
"""
This is a very simple Python program that prints "Hello".
That's all it does.
"""


print("Hello")
```

LISTS

Python also has lists (called arrays in some languages) which are collections of data of any type:

```
numbers = [1, 2, 3]
```

Lists are denoted by the use of square brackets [] and each item is separated by a comma.

ITERATION

Some data types are iterable, which means you can loop over the values they contain. For example a list:

```
numbers = [1, 2, 3]


for number in numbers:
print(number)
```

This takes each item in the list numbers and prints out the item:

```
1
2
3
```

Note I used the word number to denote each item. This is merely the word I chose for this - it's recommended you choose descriptive words for variables - using plurals for lists, and singular for each item makes sense. It makes it easier to understand when reading.

Other data types are iterable, for example the string:

```
dog_name = "BINGO"


for char in dog_name:
print(char)
```

This loops over each character and prints them out:

```
B
I
N
G
O
```

RANGE

The integer data type is not iterable and trying to iterate over it will produce an error. For example:

```
fori in 3:
print(i)
```

will produce:

```
TypeError: 'int' object is not iterable
```

However you can make an iterable object using the range function:

```
fori in range(3):
print(i)
```

range(5) contains the numbers 0, 1, 2, 3 and 4 (five numbers in total). To get the numbers 1 to 5 use range(1, 6).

LENGTH

You can use functions like len to find the length of a string or a list:

```
name = "Jamie"
print(len(name))  # 5


names = ["Bob", "Jane", "James", "Alice"]
print(len(names))  # 4
```

IF STATEMENTS

You can use if statements for control flow:

```
name = "Joe"


iflen(name) > 3:
```

```
print("Nice name,")
print(name)
else:
print("That's a short name,")
print(name)
```

PYTHON FILES IN IDLE

To create a Python file in IDLE, click File > New File and you'll be given a blank window. This is an empty file, not a Python prompt. You write a Python file in this window, save it, then run it and you'll see the output in the other window.

For example, in the new window, type:

```
n = 0


fori in range(1, 101):
   n += i


print("The sum of the numbers 1 to 100 is:")
print(n)
```

Then save this file (File > Save or Ctrl + S) and run (Run > Run Moduleor hit F5) and you'll see the output in your original Python window.

EXECUTING PYTHON FILES FROM THE COMMAND LINE

You can write a Python file in a standard editor like Vim, Nano or LeafPad, and run it as a Python script from the command line. Just navigate to the directory the file is saved (use cd and ls for guidance) and run with python, e.g. python hello.py.

TENSORFLOW-INTRODUCTION

TensorFlow is a software library or framework, designed by the Google team to implement machine learning and deep learning concepts in the easiest manner. It combines the computational algebra of optimization techniques for easy calculation of many mathematical expressions. The official website of TensorFlow is mentioned below: https://www.tensorflow.org/

Let us now consider the following important features of TensorFlow:

- It includes a feature of that defines, optimizes and calculates mathematical expressions easily with the help of multi-dimensional arrays called tensors.
- It includes a programming support of deep neural networks and machine learning techniques.
- It includes a high scalable feature of computation with various data sets.
- TensorFlow uses GPU computing, automating management. It also includes a unique feature of optimization of same memory and the data used.

Why is TensorFlow So Popular?

TensorFlow is well-documented and includes plenty of machine learning libraries. It offers a few important functionalities and methods for the same. TensorFlow is also called a "Google" product. It includes a variety of machine learning and deep learning algorithms. TensorFlow can train and run deep neural networks for handwritten digit classification, image recognition, word embedding and creation of various sequence models.

TensorFlow — Installation

To install TensorFlow, it is important to have "Python" installed in your system. Python version 3.4+ is considered the best to start with TensorFlow installation. Consider the following steps to install TensorFlow in Windows operating system.

```
pip install tensorflow
```

TensorFlow — Convolutional Neural Networks

After understanding machine-learning concepts, we can now shift our focus to deep learning concepts. Deep learning is a division of machine learning and is considered as a crucial step taken by researchers in recent decades. The examples of deep learning implementation include applications like image recognition and speech recognition.

Following are the two important types of deep neural networks:

- Convolutional Neural Networks
- Recurrent Neural Networks In this chapter, we will focus on the CNN, Convolutional Neural Networks

Convolutional Neural Networks

Convolutional Neural networks are designed to process data through multiple layers of arrays. This type of neural networks is used in applications like image recognition or face recognition. The primary difference between CNN and any other ordinary neural network is that CNN takes input as a two-dimensional array and operates directly on the images rather than focusing on feature extraction which other neural networks focus on. The dominant approach of CNN includes solutions for problems of recognition. Top companies like Google and Facebook have invested in research and development towards recognition projects to get activities done with greater speed.

A convolutional neural network uses three basic ideas:

- Local respective fields
- Convolution
- Pooling

Let us understand these ideas in detail.

CNN utilizes spatial correlations that exist within the input data. Each concurrent layer of a neural network connects some input neurons. This specific region is called local receptive field. Local receptive field focusses on the hidden neurons. The hidden neurons process the input data inside the mentioned field not realizing the changes outside the specific boundary.

If we observe the above representation, each connection learns a weight of the hidden neuron with an associated connection with movement from one layer to another. Here, individual neurons perform a shift from time to time. This process is called "convolution". The mapping of connections from the input layer to the hidden feature map is defined as "shared weights" and bias included is called "shared bias". CNN or convolutional neural networks use pooling layers, which are the layers, positioned immediately after CNN declaration. It takes the input from the user as a feature map that comes out of convolutional networks and prepares a condensed feature map. Pooling layers helps in creating layers with neurons of previous layers.

KERAS

INTRODUCTION

Deep learning is one of the major subfield of machine learning framework. Machine learning is the study of design of algorithms, inspired from the model of human brain. Deep learning is becoming more popular in data science fields like robotics, artificial intelligence(AI), audio & video recognition and image recognition. Artificial neural network is the core of deep learning methodologies. Deep learning is supported by various libraries such as Theano, TensorFlow, Caffe, Mxnet etc., Keras is one of the most powerful and easy to use python library, which is built on top of popular deep learning libraries like TensorFlow, Theano, etc., for creating deep learning models.

OVERVIEW OF KERAS

Keras runs on top of open source machine libraries like TensorFlow, Theano or Cognitive Toolkit (CNTK). Theano is a python library used for fast numerical computation tasks. TensorFlow is the most famous symbolic math library used for creating neural networks and deep learning models. TensorFlow is very flexible and the primary benefit is distributed computing. CNTK is deep learning framework developed by Microsoft. It uses libraries such as Python, C#, C++ or standalone machine learning toolkits. Theano and TensorFlow are very powerful libraries but difficult to understand for creating neural networks. Keras is based on minimal structure that provides a clean and easy way to create deep learning models based on TensorFlow or Theano. Keras is designed to quickly define deep learning models. Well, Keras is an optimal choice for deep learning applications.

FEATURES

Keras leverages various optimization techniques to make high level neural network API easier and more performant. It supports the following features:

- Consistent, simple and extensible API.
- Minimal structure - easy to achieve the result without any frills.
- It supports multiple platforms and back ends.
- It is user friendly framework which runs on both CPU and GPU.
- Highly scalability of computation.

BENEFITS

Keras is highly powerful and dynamic framework and comes up with the following advantages:

- Larger community support.
- Easy to test.
- Keras neural networks are written in Python which makes things simpler.
- Keras supports both convolution and recurrent networks.
- Deep learning models are discrete components, so that, you can combine into many ways.

KERAS ― OVERVIEW OF DEEP LEARNING

Deep learning is an evolving sub field of machine learning. Deep learning involves analyzing the input in layer by layer manner, where each layer progressively extracts higher level information about the input. Let us take a simple scenario of analyzing an image. Let us assume that your input image is divided up into a rectangular grid of pixels. Now, the first layer abstracts the pixels. The second layer understands the edges in the image. The Next layer constructs nodes from the edges. Then, the next would find branches from the nodes. Finally, the output layer will detect the full object. Here, the feature extraction process goes from the output of one layer into the input of the next subsequent layer. By using this approach, we can process huge amount of features, which makes deep learning a very powerful tool. Deep learning algorithms are also useful for the analysis of unstructured data. Let us go through the basics of deep learning in this chapter.

CODING:
TESTING MODEL:

```
# -*- coding: utf-8 -*-
# MLP for Pima Indians Dataset Serialize to JSON and HDF5
import numpy as np
from keras.preprocessing import image
from keras.models import Sequential
from keras.layers import Dense
from keras.models import model_from_json
import os
import cv2

json_file = open('model1.json', 'r')
loaded_model_json = json_file.read()
json_file.close()
loaded_model = model_from_json(loaded_model_json)
# load weights into new model
loaded_model.load_weights("model1.h5")
print("Loaded model from disk")

##label=['burger','chicken
briyani','dosa','idly','pizza','pongal','poori','white rice']

def classify(img_file):
    img_name = img_file
    test_image = image.load_img('data/test/burger_4.jpeg', target_size =
(128, 128))

    test_image = image.img_to_array(test_image)
    test_image = np.expand_dims(test_image, axis=0)
    result = model.predict(test_image)
    a=np.round(result[0][0])
    b=np.round(result[0][1])
    c=np.round(result[0][2])
    d=np.round(result[0][3])
    a=np.round(result[0][4])
    b=np.round(result[0][5])
    c=np.round(result[0][6])
    d=np.round(result[0][7])

    print(a)
    print(b)
    print(c)
    print(d)
    print(e)
    print(f)
    print(g)
    print(h)
    if a == 1:
        prediction = 'burger'
        print(prediction)
    elif b == 1:
        prediction = 'chicken briyani'
        print(prediction)
    elif c == 1:
        prediction = 'dosa'
        print(prediction)
    elif d  == 1:
        prediction = 'idly'
        print(prediction)
    elif e == 1:
        prediction = 'pizza'
        print(prediction)
    elif f == 1:
        prediction = 'pongal'
        print(prediction)
    elif g == 1:
        prediction = 'poori'
        print(prediction)
    elif h  == 1:
        prediction = 'white rice'
        print(prediction)



##test_image = image.img_to_array(test_image)
##test_image = np.expand_dims(test_image, axis = 0)
##result = loaded_model.predict(test_image)
##print(result)
##fresult=np.max(result)
##label2=label[result.argmax()]
###print(label2)
```

l’RAINING MODEL:

```
# -*- coding: utf-8 -*-
from keras.models import Sequential
#initialize nn

from keras.layers import Conv2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.preprocessing.image import ImageDataGenerator
from keras.layers import Dense

from keras.layers import BatchNormalization
from keras.layers import Dropout

#basic cnn
model = Sequential()
model.add(Conv2D(32, kernel_size = (3, 3), activation='relu',
input_shape=(128,128, 3)))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(BatchNormalization())
model.add(Conv2D(64, kernel_size=(3,3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(BatchNormalization())
model.add(Conv2D(64, kernel_size=(3,3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(BatchNormalization())
model.add(Conv2D(96, kernel_size=(3,3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(BatchNormalization())
model.add(Conv2D(32, kernel_size=(3,3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(BatchNormalization())
model.add(Dropout(0.2))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.3))
model.add(Dense(8, activation = 'softmax'))

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



train_datagen = ImageDataGenerator(rescale = None,
                                   shear_range = 0.2,
                                   zoom_range = 0.2,
                                   horizontal_flip = True)

test_datagen = ImageDataGenerator(rescale = 1./255)

training_set = train_datagen.flow_from_directory('data/train',
                                                 target_size = (128, 128),
                                                 batch_size = 8,
                                                 class_mode =
'categorical')
#print(test_datagen);
labels = (training_set.class_indices)
print(labels)

test_set = test_datagen.flow_from_directory('data/val',
                                            target_size = (128, 128),
                                            batch_size = 8,
                                            class_mode = 'categorical')

labels2 = (test_set.class_indices)
print(labels2)

model.fit_generator(training_set,
                         steps_per_epoch = 50,
                         epochs = 100,
                         validation_data = test_set,
                         validation_steps = 100)



model_json=model.to_json()
with open("model1.json", "w") as json_file:
    json_file.write(model_json)
# serialize weights to HDF5
    model.save_weights("model1.h5")
    print("Saved model to disk")
```

RESULI':

This section discusses the results, and the observations found while experimenting starting from the performance measurement techniques for food classification.

The performance of the system is high, and is considered acceptable from a usage point of view. However, the CNN need high-performance computing machines in order to experiment on the huge multi-media datasets. The CNN is capable of train highly non-linear data, and for that in contrast, it takes more computational time to train the network.However, the performance matters a lot, and once the system is properly trained, the system can produce the results in less time. The images are properly pre processed and all kinds of images are tested with CNN. From this, it is concluded that CNN are more suitable for classifying the images when the number of classes are more.The task of image classification can be extended using prominent features that can categorize food images. Since the CNN are consuming high computational time, the feature-based approach is highly appreciable. A multi-level classification approach (hierarchical approach) is suitable to avoid mis-classifications when the number of classes is more.Moreover, a dataset containing all food categories is also not available in the literature yet.

CONCLUSION

Food plays an essential role in human life, providing various nutrients, and therefore the consumption of food is crucial for our health. Food classification is therefore a crucial aspect in maintaining a healthy lifestyle. In the world of health and medicine, food image classification is an emerging research area. A survey of automatic food classification methods based on Convolutional Neural Networks has been presented. The majority of the work uses the Food-101 dataset to train the models. Among the different approaches, InceptionV3-based systems provide higher accuracy in food image classification.

ADVANTAGE :

- Opportunities to help people understand there daily eating habits, nutricians patterns maintain a healthy exploring diet.