

UAT INITIATION AND DESIGN

1. Machine Learning

It is essential to choose the right strategy; Machine-learning techniques are often used to do this. Machine learning is part of artificial intelligence (AI). It can be defined as an algorithm that focuses on computer program development [14, 15, 35, 43]. Machine learning considered an application that use to increase computer ability to learn from previous experiences [16, 17, 44].

There are three types of ML:

- Supervised:

The application has previous experience of new data using labelled examples to predict future events [18, 19].

- Semi-supervised:

Data can be labelled and unlabelled for training [29, 36].

- Unsupervised:

Data that unlabelled from experience. The system cannot figure out the right input so that the result will be wrong [20, 21, 37].

2. Deep Learning

Compared to standard algorithms, neural networks can solve somewhat complicated issues at a much easier level about the complexity of algorithms. Neural networks can solve somewhat complicated issues at a much easier level concerning the complexity of algorithms [26, 30]. The neural network builds to mimic human brain neural function but with the mathematical functions [31, 33,38]. One of the neural networks is the multi-layer network as shown in figure 1. It includes three layers, the input layer, many hidden layers, and the output layer [21,39]. The input layer passes data without modification. Hidden layers process the data, and the output layer converts hidden layers to output as a classification. Collecting datasets for training takes time to process [41, 45]. As the number of configuration increases, training samples increase.

3. Decision Tree

A decision tree uses datasets that include attribute vectors that contain a collection of classification class [16, 20]. This technique builds the dataset on attributes that split the data to different existing classes until the criterion is reached [20, 31]. The decision tree has an excellent representation form to have an idea of data quickly due to easy visualization format. The structure of the decision tree contains nodes that separated into root node, inner node, and end node 'leaf'. The root node is the start of the decision has no incoming edges [32, 40]. Inner nodes include one incoming edge and at least more than one outgoing edge. A decision tree is a tool that uses most of the time to support decisions because it shows the rule of sequential decisions [16].

4. Image Recognition Process

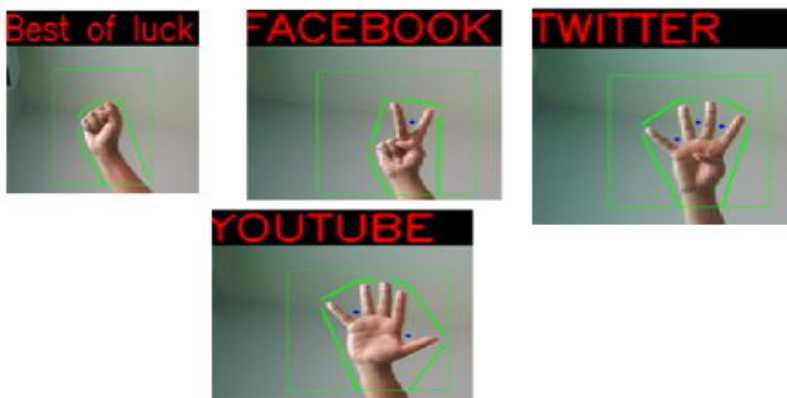
The image recognition process is a process that enables the input of the sign language into the application for necessary processing [20,31,46]. The process requires a sign to be made in front of the webcam. The computer captures the sign made via the webcam and stores the different images made. Images that come from the camera will be resized, and the resolution will change. The colors will change to grayscale image and then to black and white images while editing the images [25, 33,47]. There several techniques used to extract the image, such as SIFT, SURF, BRISK, and HSV algorithms. Scale Invariant Feature Transform "SIFT" is used to extract features vectors that define local patches of the image [35]. These features are invariant to translation, rotation, and illumination, not only vectors scale-invariant. There are several advantages to the SIFT descriptor [36]. For example, this descriptor is accurate than any other descriptor. It can describe the key points in the image for any object. It allows an individual feature to define the correct match with the best probability in a huge database of features [37]. Also, it can cover the full range scales and location of the image, and it is close to real-time performance.

UAT EXECUTION

In the implementation phase, developers change several tasks that they were planned to do. They notice that they can build the system without preparing any training and testing images as they were plan. The code is depending on skin color and contour to find the right sign. Moreover, developers narrow the tasks to only one task which is browse websites only. Moreover, the result was precise and accurate aligned with the methodology and testing that was used. This signifies that developing modern technology assists disabled individuals specifically deaf-dumb on interacting among people. The measurement variables along with the supporting evidence from the methodology concluded that the measures taken to evaluate this study were supported all throughout. This system will recognize any element in the box, and the brightness does not matter. D-talk is a dynamic system that includes three gestures in total to browse websites. All that users need to implement this system is WiFi connection and webcam to capture user gestures. The following figures 10 and 11 show the hand gestures that are used for orders that the system can recognize to browse websites. It could be used as a guide for users.



Figure 9: Application main screen



UTILIZATION OF TESTING TOOLS

IBM WATSON CLOUD

IBM Cloud Parks give developers, data managers and administrators an open environment to quickly build new cloud-native applications, modernize existing applications, and extend the AI capabilities of IBM Watson into their business in a consistent manner across multiple clouds.

Watson helps organizations predict future outcomes, automate complex processes, and optimize employees' time.

IBM FINAL CODE :

1.]IBM DEPLOYMENT

```
!pip install watson-machine-learning-client  
  
from ibm_watson_machine_learning import APIClient  
  
wml_credentials={  
    "url":"https://us-south.ml.cloud.ibm.com",  
    "apikey":"x91CJTUTrrIfLvrXsKf8yLyl1KHb3JV0Y7Qrwy1zilb2"  
}  
  
client=APIClient(wml_credentials)
```

2.]CLIENT

```
def guid_space_name(client,animal_deploy):  
    space=client.spaces.get_details()  
  
    return(next(item for item in space['resources'] if  
    item['entity']['name']= animal_deploy)["metadata"]["id"])  
  
space_uid=guid_space_name(client,'animal_deploy')  
  
print("Space UID "+space_uid)  
  
client.set.default_space(space_uid)
```

```
client,software specifications.list(200)

software_space_uid=client.software_specifications.get_uid
_by_name('tensorflow_rt22.1-py3.9')

software_space_uid

model_details=client.repository.store_model(model='Dataset.tgz',meta_props={

client.repository.ModelMetaNames.NAME: "CNN Model
Building",

client.repository.ModelMetaNames.TYPE: 'tensorflow_2.7',

client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:
software_spec_uid

})

model_id=client.repo
```