Ain Shams University

Faculty of Computer & Information Sciences
Computer Science Department

Interactive Statues

Ain Shams University

Faculty of Computer & Information Sciences
Computer Science Department

Interactive Statues

By:

Essam El-din Sharif Gab-Allah Hassan	[Computer Science]
Abdallah Ramdan Elsaid Ibrahim	[Computer Science]
Hassan Walid Hassan Elsaid	[Computer Science]
Amr Mahmoud Zakaria Gad	[Computer Science]
Omar Hany Abd-El Fatah Said	[Computer Science]

Under Supervision of:

Dr. Ayat Mohammed Naguib
Scientific Computing Department,
Faculty of Computer and Information Sciences,
Ain Shams University.

TA. Zeina Rayan
Computer Science Department,
Faculty of Computer and Information Sciences,
Ain Shams University.

TA. Mohammad Essam
Bioinformatics Department,
Faculty of Computer and Information Sciences,
Ain Shams University.

Acknowledgment

First and foremost, we want to express our gratitude to God for his blessings on success in this project.

The team members put a lot of effort into finishing the project, and we all tried our best to make it the best we could.

We thank our project supervisor, Dr. Ayat Mohammed Naguib, for her advice. Her suggestions and guidance were helpful in getting our project finished.

We also want to extend my gratitude to my moderators, TA. Zineb Rayan and TA. Mohamed Essam who supported us all the time Our thesis project.

Finally, we want to thank our families, friends, and colleagues for supporting and helping us.

Abstract

When you go to the museum and want to know information about a specific statue from the many statues in this museum, it is difficult to provide a tour guide for each person for free, so we made this application to help everyone who wants to know information about a specific statue in a different and easy way.

This application plays the role of a tour guide by opening a conversation with the statue that the user wants information about this statue, and the user asks a question about this statue, so the application responds to this question through a fake video in which the statue appears talking about the answer to this question.

So, in this documentation, we will explain the techniques we used to identify the statues and create the fake videos for each statue, and we will explain the final result that we have reached for this application, which can identify statues and answer each question through a fake video of this statue that talks about the answer to this question.

It was difficult to have a ready-made data set for our project, so we collected a data set for our project ourselves, and we made sure of the validity of this information that we collected for each statue through doctors and professors from the Faculty of Archeology.

Table Of Contents

List of Figures

List of Tables

List of Abbreviations

NN: neural network.

CNN: Convolutional Neural Network.

RNN: Recurrent neural network

DCNN: Deep convolutional neural network.

NLP: Natural language processing

AI: Artificial Intelligence

ML: machine learning

LSTMs: Long Short Term Memory Networks

MLPs: Multilayer Perceptrons

RELU: Rectified Linear Units

SVM: Support Vector Machine

GANs: Generative Adversarial Networks

VS Code: Visual Studio Code

1) Chapter 1: Introduction

Introduction

In 2021, eight million tourists from various countries visited Egypt, according to the Central Agency for Public Mobilization and Statistics recently as shown in figure 1.2 [1].

On the other hand, the captain of tour guides announced that there are only 17,000 guides, including 8,000 guides who speak English, so visitors feel annoyed because of waiting several times until they get a tour guide who knows them about the achievements and the history of the artifacts as shown in figure 1.1.

Figure 1.1 Introduction

There is another group that is disturbed by the exaggeration by the tourism companies in the prices of guided tours because of their limited number, considering that the tourist wants to visit all the available places in the country on a low budget.

So, we decided to create a mobile app that will help visitors learn information about these statues interactively and in a simplified manner. we will replace the **manual guiding method** with the **automated guiding method**.

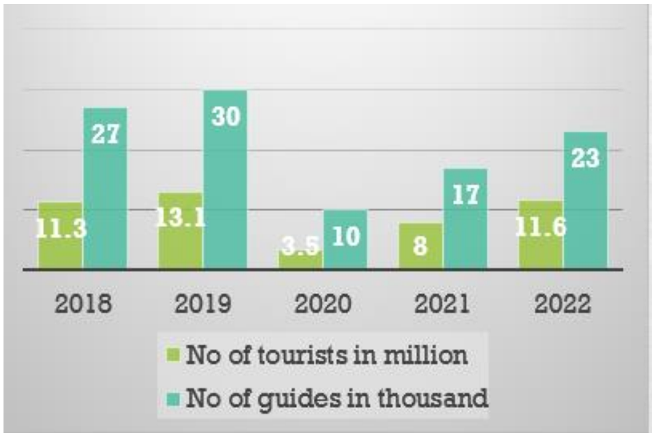

Figure 1.2: Statistics between no of tourists & tour guides

1.1 Motivation

This project aims to help a large group of users interested in visiting museums and archaeological sites (whether they are students, tourists, or researchers) and help them know the existing antiques in a simple and interactive way and know all their history and origins through an interaction between the user and the statue so that the statue speaks about itself and its history.

This project also helps fuel tourism in Egypt, expand people's knowledge of our ancient civilization, and make tourists happier during their trip while reducing the need to carry printed brochures or obtain a tour guide explaining the statues and their history. And visit more tourist places because the application will save a lot of guide costs. According to the statistics, the number of tourists is gradually increasing noticeably, and thus the need for the application and its use will increase.

1.2 Problem Definition

The problem consists that not every tourist who wants to know information about the statue has a tour guide, so we thought of using deepfake technology to solve this problem by making a fake video for each statue talking about himself in it, knowing that deepfake technology is used in a large percentage only for fun and entertainment, So we thought of using it as something useful and as a tour guide for tourists.

1.3 Objective

- Create efficient user-friendly android tour-guide application for tourists and citizens.
- The application is supposed to detect the statue through a mobile camera and recognize it.
- The core objective of the application, after statue recognition, is to create an animation video of the statue representing itself.
- The application should help the tourist to know everything about the history of monuments and statues in an interesting way.
- The application may perform some tasks of the tour guide present with the tourist who provides him information about the antiquities.
- It is possible to dispense the tour guide present with the tourist and reduce the cost for him.

1.4 Methodology

We utilized a convolutional neural network (CNN) to classify and identify images of statues that were input into the application via a phone's camera or photo gallery. we experimented with various CNN architectures, including VGG16, VGG19, ResNet, Inception, GoogLeNet, AlexNet, and a custom CNN. Then we concluded that the VGG-19 architecture was the best model, considering factors such as results, speed, and accuracy.

We use deep fake models in our project to create a fake video of the statue in which he speaks and gives information about himself, We tried many deep fake models such as First Order Motion Model, Make It Talk Model and Face2Face and we found that the best model to create fake videos of statues in our project is First Order Motion Model according to results, speed, and accuracy.

The project involves the use of a voice analysis algorithm to analyze the questions that users ask and provide responses from the virtual statue accordingly. The system includes a Conversation page where users can submit questions either by recording them or selecting from a list of predefined questions. Once a question is submitted, the system returns a fake video containing the answer from the virtual statue.

1.5 Time Plan

Steps of Time Plan as shown in figure 1.3:

- Learning process.
- Research, Survey, and Collect the data set.
- Preprocessing our data.
- Create our models.
- Create a mobile application and design a database.
- Testing and integration of our models.

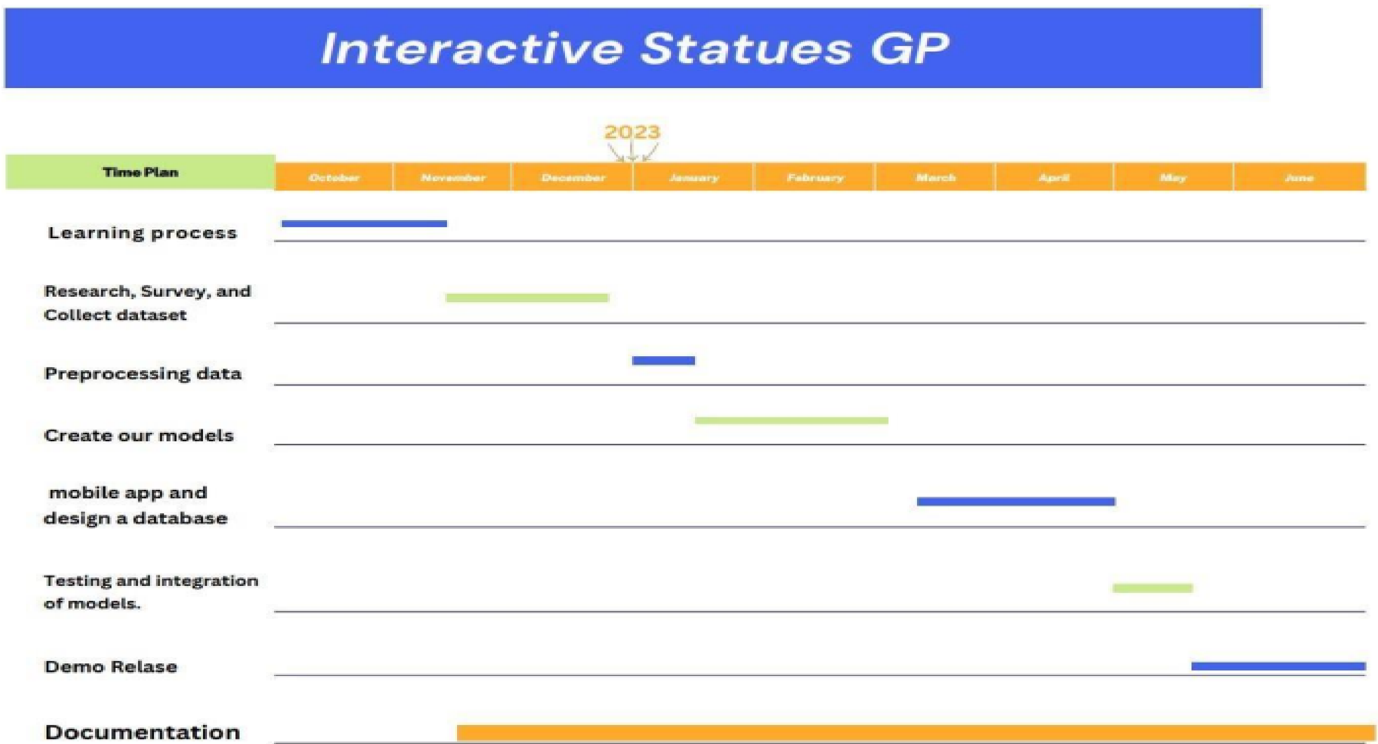

Figure 1.3 Time Plan

1.6 Thesis Outline:

Chapter 2: Literature Review

the contents of a section that deals with the field of the project - deep learning and deepfake - and discusses its scientific background, analysis, and surveys. The section aims to provide a thorough and comprehensive overview of the field, including its history, development, and current state. Additionally, the section presents the results of surveys and studies conducted on the subject, highlighting key findings and insights.

Chapter 3: System Architecture and Methods

This section focuses on presenting the details of the system architecture and its main modules. It aims to provide readers with a complete picture of the technical aspects of the project. Additionally, the detailed description of the methods and procedures used to develop and implement the system will also be presented in this section.

Chapter 4: System Implementation and Results

The section will offer a detailed description of the dataset that was used for the project, as well as the software tools that were utilized in the project. In addition, the section will cover the configuration steps that were involved in the application, design of experiments, and the results that were obtained as part of the project. The aim is to provide readers with a comprehensive understanding of how the dataset was utilized to develop the project, and how the results were obtained. The section will give a complete overview of the technical processes that were involved in the project, helping readers to understand the methodology employed in the project development process.

Chapter 5: Run the Application

This section includes a step-by-step guide for users to learn how to use the desktop application and the user manual for the mobile application.

Chapter 6: Conclusion and Future work

The section provides a complete summary of the project, including the results that were obtained. It will also offer insight into how the project's performance could be improved in the future and what additional features or functions could be added to enhance its capabilities.

2) Chapter 2: Literature Review

1-Analysis and comparison our study

we make surveys to save time. They are relatively inexpensive and useful in our project. Surveys can have a very small cost per participant and a high number of potential responses. Gathering results can be very quick and one way to ensure success in our project is to utilize techniques obtained from surveys [2].

we make surveys to study the best techniques used, we can ensure that the information we obtain from surveys will be accurate, with such valuable data in hand, we can make informed decisions on how to proceed with our project, ultimately achieving the best results possible.

Comparison of our surveys

In the paper "A large -scale Challenging Dataset for deepfake Forensics" in this paper used Convolutional neural network (CNN) and FaceForensics as dataset and the study yielded results, achieving an accuracy rate of 85.0% as shown in table 1 [3].

In the paper "Make It Talk: Speaker-Aware Talking-Head Animation" in this paper used Make It Talk Model and VoxCeleb2 as dataset and the study yielded results, achieving an accuracy is rate of 88.0% as shown in table 1 [4].

In the paper titled "Deep Learning for Deepfakes Creation and Detection" investigates the use of VGG16-ResNet50 models on the CelebA dataset for deepfake creation and detection. The study yielded promising results, with the model achieving an accuracy rate of 83.0% as shown in table 1 [5].

In the paper "First order motion Model for image Animation" describes a study that investigates the effectiveness of the first-order motion model and X2Face and Monkey-Net techniques for image animation using VoxCeleb 1, Tai-Chi-HD, as datasets. the study yielded promising results, achieving an accuracy is rate of 68.4% when use Monkey-Net technique and VoxCeleb 1 as dataset, achieving an accuracy is rate of 80.6% when use Monkey-Net technique and Tai-Chi-HD as dataset, achieving an accuracy is rate of 88% when use X2Face technique and Tai-Chi-HD as dataset, achieving an accuracy is rate of 90.8% when use X2Face technique and VoxCeleb 1 as dataset as shown in table 1 [6].

In the paper titled " Study of Face Recognition Techniques" investigates the use of MS-HMM and NN Based SOM for Face recognition techniques on the IIT-Dehli and UMIST as dataset for face recognition. The study yielded promising results, achieving an accuracy rate of 85.25% when using NN Based SOM for Face recognition technique and IIT-Dehli as dataset and achieving an accuracy rate of 90.66% when using MS-HMM technique and UMIST as dataset as shown in table 2 [7].

In the paper "Convolutional Neural Network CNN for Image Detection and Recognition" details a study that explores the effectiveness of CNN and RNN models when applied to image detection and recognition using the CIFAR-10 dataset. The results showed that the CNN model had a higher accuracy rate of 80.17%, while the RNN model achieved a slightly lower accuracy rate of 78.90%. as shown in table 2 [8].

In the paper "Data-specific Adaptive Threshold for Face Recognition" details a study that explores the effectiveness of DCNN model when applied to image recognition using the Adience dataset. The study yielded promising results, achieving an accuracy rate of 84.30% as shown in table 2 [9].

In the paper "A Study on CNN Transfer Learning" details a study that explores the effectiveness of CNN model when applied to image recognition using the Caltech Faces dataset. The study yielded promising results, achieving an accuracy rate of 92.85% as shown in table 2 [10].

Summary of table 1 the best model used based on high accuracy is the first order motion model to create fake video that is used in the paper “First Order Motion Model for Image Animation” on VoxCeleb 1 dataset.

Summary of table 2 the best model used based on high accuracy is CNN model that is used in the paper “A Study on CNN Transfer Learning” on Caltech Faces dataset.

Table 2.1 : Deep Fake Survey

Paper Name	Model / Tech	Data set	Accuracy
First Order Motion Model for Image Animation	First Order Motion / X2Face	VoxCeleb 1	90.8%
		Tai-Chi-HD	88.0%
	First Order Motion / Monkey-Net	VoxCeleb 1	68.4%
		Tai-Chi-HD	80.6%
A large -scale Challenging Dataset for deepfake Forensics	Convolutional neural network (CNN)	FaceForensics	85%
Make It Talk: Speaker-Aware Talking-Head Animation	Make It Talk Model	VoxCeleb2	88%
Deep Learning for Deepfakes Creation and Detection	VGG16 – ResNet50	5,000 real images from CelebA and 5,000 fake images.	83.3%

Table 2.2 : Deep Learning Survey

Paper Name	Model / Tech	Data set	Accuracy
Study of Face Recognition Techniques	MS-HMM	UMIST	90.66%
	NN Based SOM for Face recognition.	IIT-Dehli	85.25%
Convolutional Neural Network CNN for Image Detection and Recognition	CNN	CIFAR-10	80.17%
	RNN		78.90%
A Study on CNN Transfer Learning	CNN	Caltech Faces	92.85%
Data-specific Adaptive Threshold for Face Recognition	DCNN	Adience 19,339 Images 2,284 Classes	84.30%

2-Background

2.1 Artificial Intelligence:

2.1.1 What Is Artificial Intelligence (AI)?

Artificial intelligence (AI) refers to the simulation of human intelligence in machines that are programmed to think like humans and mimic their actions. The term may also be applied to any machine that exhibits traits associated with a human mind such as learning and problem-solving.

2.1.2 Applications for artificial

The applications for artificial intelligence are endless. The technology can be applied to many different sectors and industries.AI is being tested and used in the healthcare industry for dosing drugs and doling out different treatments tailored to specific patients, and for aiding in surgical procedures in the operating room.

2.1.3 Types of Artificial Intelligence

Artificial intelligence can be divided into two different categories: weak and strong. **Weak artificial intelligence** embodies a system designed to carry out one job. Weak AI systems include video games such as the chess example from above and personal assistants such as Amazon's Alexa and Apple's Siri. You ask the assistant a question, and it answers it for you.

Strong artificial intelligence systems are systems that carry on the tasks considered to be human-like. These tend to be more complex and complicated systems. They are programmed to handle situations in which they may be required to problem solve without having a person intervene. These kinds of systems can be found in applications like self-driving cars or in hospital operating rooms [11].

2.1.4 How does AI work?

To begin with, an AI system accepts data input in the form of speech, text, image, etc. The system then processes data by applying various rules and algorithms, interpreting, predicting, and acting on the input data. Upon processing, the system provides an outcome, i.e., success or failure, on data input. The result is then assessed through analysis, discovery, and feedback. Lastly, the system uses its assessments to adjust input data, rules and algorithms, and target outcomes. This loop continues until the desired result is achieved as presented in figure 2.1 **[12]**.

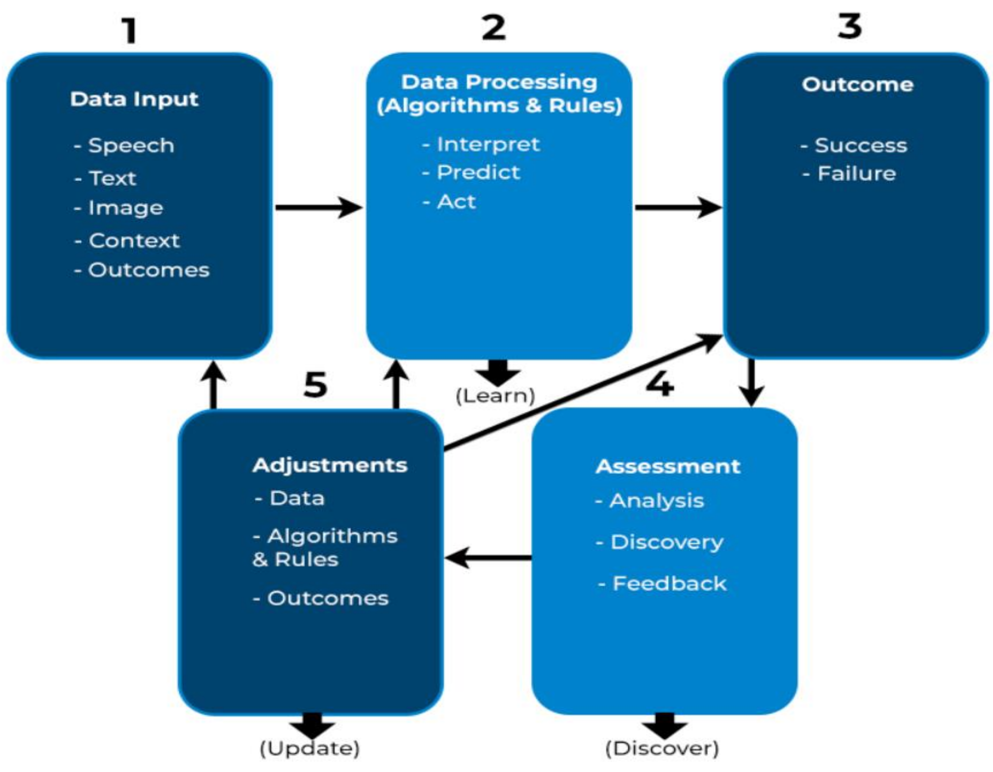

Figure 2.1: AI work

2.1.5 Key Components of AI

The key components of Artificial Intelligence (AI) consists of Machine learning, Deep learning, Neural network, Cognitive computing, Natural language processing and Computer vision as shown in figure 2.2.

Machine learning: Machine learning is an AI application that automatically learns and improves from previous sets of experiences without the requirement for explicit programming.

Deep learning: Deep learning is a subset of ML that learns by processing data with the help of artificial neural networks.

Neural network: Neural networks are computer systems that are loosely modeled on neural connections in the human brain and enable deep learning.

Cognitive computing: Cognitive computing aims to recreate the human thought process in a computer model. It seeks to imitate and improve the interaction between humans and machines by understanding human language and the meaning of images.

Natural language processing (NLP): NLP is a tool that allows computers to comprehend, recognize, interpret, and produce human language and speech.

Computer vision: Computer vision employs deep learning and pattern identification to interpret image content (graphs, tables, PDF pictures, and videos) [3].

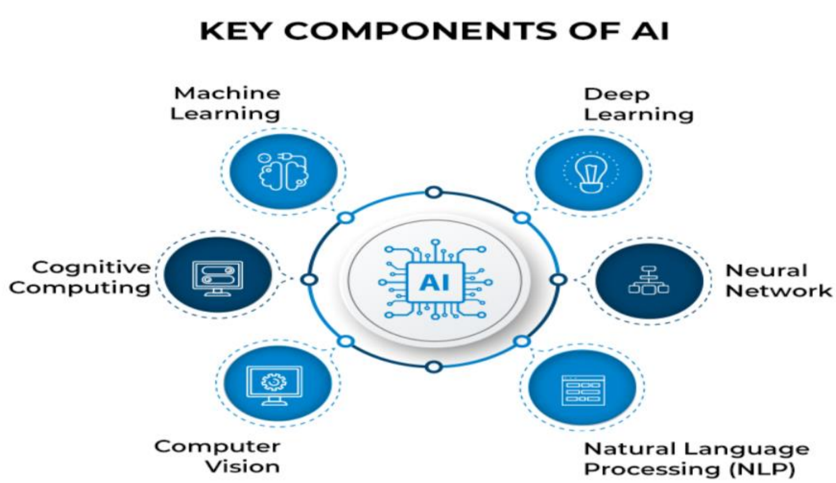

Figure 2.2: Key components of AI

2.2 Machine Learning:

2.2.1 What Is Machine Learning?

Machine learning is an application of AI that enables systems to learn and improve from experience without being explicitly programmed. Machine learning focuses on developing computer programs that can access data and use it to learn for themselves. [13]

Machine learning methods enable computers to operate autonomously without explicit programming. ML applications are fed with new data, and they can independently learn, grow, develop, and adapt [14] [15].

2.2.2 How does machine learning work?

Machine learning algorithms are molded on a training dataset to create a model. As new input data is introduced to the trained ML algorithm, it uses the developed model to make a prediction.

Further, the prediction is checked for accuracy. Based on its accuracy, the ML algorithm is either deployed or trained repeatedly with an augmented training dataset until the desired accuracy is achieved as shown in figure 2.3 [14].

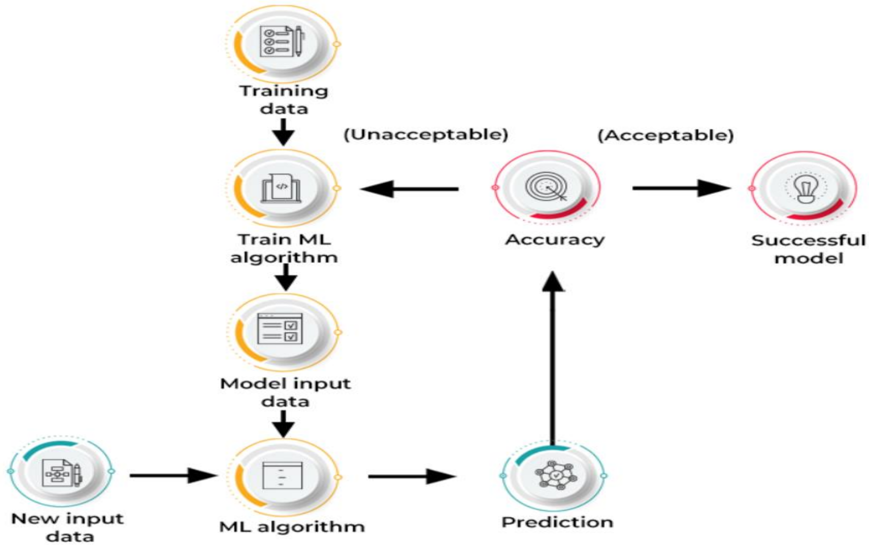

Figure 2.3: How does machine learning work

2.2.3 Classification of Machine Learning

The key components of Classification of Machine Learning consist of Supervised learning, Deep learn, Unsupervised learning, and Reinforcement learning as shown in figure 2.4.

- **Supervised learning:**

  Supervised learning is a type of machine learning method in which we provide sample labeled data to the machine learning system to train it, and on that basis, it predicts the output.

  The goal of supervised learning is to map input data with the output data. Supervised learning is based on supervision, and it is the same as when a student learns things in the supervision of the teacher. An example of supervised learning is spam filtering.

  Supervised learning can be grouped further into two categories of algorithms: Classification and Regression.

- **Unsupervised learning:**

  Unsupervised learning is a learning method in which a machine learns without any supervision.

  The training is provided to the machine with the set of data that has not been labeled, classified, or categorized, and the algorithm needs to act on that data without any supervision. The goal of unsupervised learning is to restructure the input data into new features or a group of objects with similar patterns.

  In unsupervised learning, we don't have a predetermined result. The machine tries to find useful insights from the huge amount of data. It can be further classifieds into two categories of algorithms: Clustering and Association.

- **Reinforcement learning:**

Reinforcement learning is a feedback-based learning method, in which a learning agent gets a reward for each right action and gets a penalty for each wrong action. The agent learns automatically with these feedbacks and improves its performance. In reinforcement learning, the agent interacts with the environment and explores it.

The goal of an agent is to get the most reward points, and hence, it improves its performance.

The robotic dog, which automatically learns the movement of his arms, is an example of Reinforcement learning [6].

Figure 2.4: Classification of Machine Learning

2.2.4 Algorithm of Machine Learning

- **Support Vector Machine Algorithm (SVM)**

What is SVM:

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

Support Vector Machines (SVM) can handle both classification and regression problems. In this method hyperplane needs to be defined which is the decision boundary. When there is a set of objects belonging to different classes then the decision plane is needed to separate them. The objects may or may not be linearly separable in which case complex mathematical functions called kernels are needed to separate the objects which are members of different classes.

SVM aims at correctly classifying the objects based on examples in the training data set.it can handle both semi structured and structured data, it can handle complex function if the appropriate kernel function can be derived. As generalization is adopted in SVM so there is less probability of over fitting. It can scale up with high dimensional data. It does not get stuck in local optima [16].

How an SVM works:

A simple linear SVM classifier works by making a straight line between two classes as shown in figure 2.5. That means all the data points on one side of the line will represent a category and the data points on the other side of the line will be put into a different category. This means there can be an infinite number of lines to choose from as shown in figure 2.6.

What makes the linear SVM algorithm better than some of the other algorithms, like k-nearest neighbors, is that it chooses the best line to classify your data points. It chooses the line that separates the data and is the furthest away from the closet data points as possible.

A 2-D example helps to make sense of all the machine learning jargon. Basically, you have some data points on a grid. You're trying to separate these data points by the category they should fit in, but you don't want to have any data in the wrong category. That means you're trying to find the line between the two closest points that keeps the other data points separated.

So, the two closest data points give you the support vectors you'll use to find that line. That line is called the decision boundary.

The decision boundary doesn't have to be a line. It's also referred to as a hyperplane because you can find the decision boundary with any number of features, not just two [17].

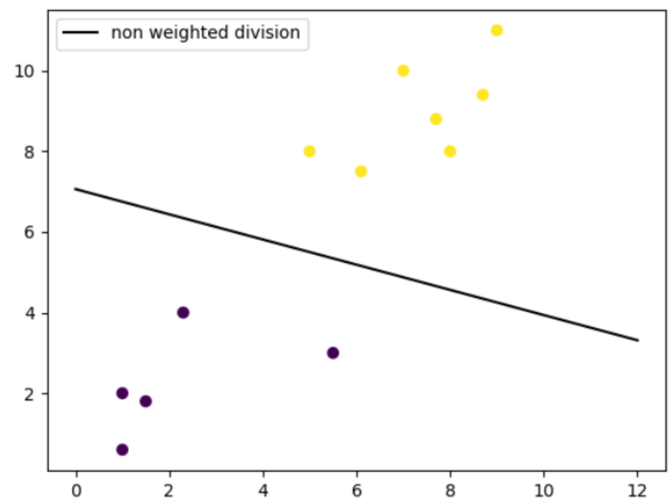

Figure 2.6:Linear SVM

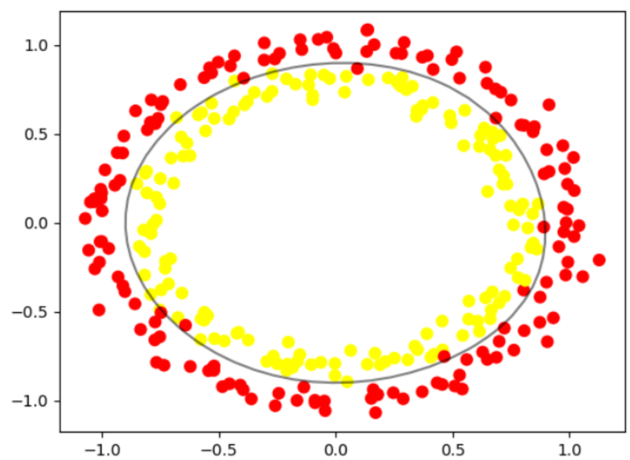

Figure 2.5:nonlinear SVM

Types of Support Vector Machines:

Support vector machines are broadly classified into two types: simple or linear SVM and kernel or non-linear SVM.

Simple or linear SVM:

A linear SVM refers to the SVM type used for classifying linearly separable data. This implies that when a dataset can be segregated into categories or classes with the help of a single straight line, it is termed a linear SVM, and the data is referred to as linearly distinct or separable. Moreover, the classifier that classifies such data is termed a linear SVM classifier.

Kernel or non-linear SVM:

Non-linear data that cannot be segregated into distinct categories with the help of a straight line is classified using a kernel or non-linear SVM. Here, the classifier is referred to as a non-linear classifier. The classification can be performed with a non-linear data type by adding features into higher dimensions rather than relying on 2D space. Here, the newly added features fit a hyperplane that helps easily separate classes or categories [18].

Examples of Support Vector Machines

Data classification:

SVMs are known to solve complex mathematical problems. However, smooth SVMs are preferred for data classification purposes, wherein smoothing techniques that reduce the data outliers and make the pattern identifiable are used.

Facial detection & expression classification

SVMs classify facial structures vs. non-facial ones. The training data uses two classes of face entity (denoted by +1) and non-face entity (denoted as -1) and n*n pixels to distinguish between face and non-face structures. Further, each pixel is analyzed, and the features from each one is extracted that denote face and non-face characters. Finally, the process creates a square decision boundary around facial structures based on pixel intensity and classifies the resultant images [18].

2.3 Deep Learning:

2.3.1 What Is Deep Learning?

Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain—albeit far from matching its ability—allowing it to "learn" from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy [19].

Deep learning works on multiple neural networks of three or more layers and attempts to simulate the behavior of the human brain. It allows statisticians to learn from large amounts of data and interpret trends [20] [21].

2.3.2 How Do Deep Learning Neural Networks Work?

- **BIOLOGICAL NEURAL NETWORKS:**

Artificial neural networks are inspired by the biological neurons found in our brains. In fact, artificial neural networks simulate some basic functionalities of biological neural network, but in a very simplified way. Let's first look at biological neural networks to derive parallels to artificial neural networks.

In short, a biological neural network consists of numerous neurons, A typical neuron consists of a cell body, dendrites, and an axon as shown figure 2.8.

Dendrites are thin structures that emerge from the cell body. An axon is a cellular extension that emerges from this cell body. Most neurons receive signals through the dendrites and send out signals along the axon.

At the majority of synapses, signals cross from the axon of one neuron to the dendrite of another. All neurons are electrically excitable due to the maintenance of voltage gradients in their membranes. If the voltage changes by a large enough amount over a short interval, the neuron generates an electrochemical pulse called an action potential. This potential travels rapidly along the axon and activates synaptic connections [12].

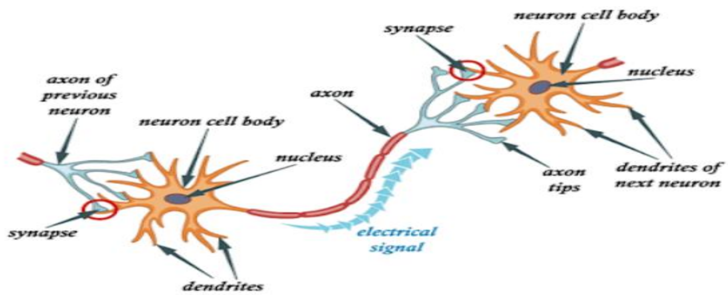

Figure 2.7: BIOLOGICAL NEURAL NETWORKS

➢ ARTIFICIAL NEURAL NETWORKS:

Now that we have a basic understanding of how biological neural networks function, let’s take a look at the architecture of the artificial neural network.

A neural network generally consists of a collection of connected units or nodes as shown in figure 2.9. We call these nodes neurons. These artificial neurons loosely model the biological neurons of our brain.

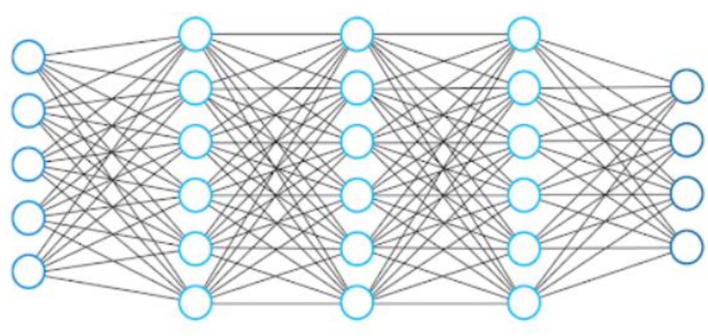

Figure 2.8: ARTIFICIAL NEURAL NETWORKS

A neuron is simply a graphical representation of a numeric value. Any connection between two artificial neurons can be considered an axon in a biological brain. The connections between the neurons are realized by so-called weights, which are also nothing more than numerical values.

When an artificial neural network learns, the weights between neurons change, as does the strength of the connection. Well, what does that mean? Given training data and a particular task such as classification of numbers, we are looking for certain set weights that allow the neural network to perform the classification.

The set of weights is different for every task and every data set. We cannot predict the values of these weights in advance, but the neural network has to learn them. The process of learning is what we call training [12].

2.3.3 Deep Learning Neural Network Architecture

The typical neural network architecture consists of several layers as shown in figure 2.10; we call the first one the input layer.

The input layer receives input x, (i.e., data from which the neural network learns). In our previous example of classifying handwritten numbers, these inputs x would represent the images of these numbers (x is basically an entire vector where each entry is a pixel).

The input layer has the same number of neurons as there are entries in vector x. In other words, each input neuron represents one element in the vector.

The last layer is called the output layer, which outputs a vector y representing the neural network's result. The entries in this vector represent the values of the neurons in the output layer. In our classification, each neuron in the last layer represents a different class.

The network must perform certain mathematical operations, which it performs in the layers between the input and output layers. We call these the hidden layers. Now let's discuss what the connections between the layers look like [12].

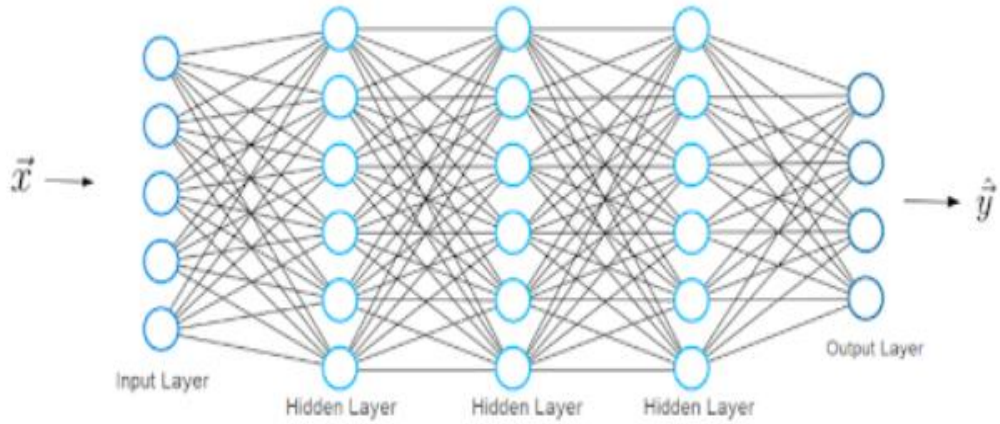

Figure 2.9: Deep Learning Neural Network Architecture

2.3.4 What's the Difference Between Machine Learning and Deep Learning?

Deep learning is a specialized form of machine learning. A machine learning workflow starts with relevant features being manually extracted from images. The features are then used to create a model that categorizes the objects in the image. With a deep learning workflow, relevant features are automatically extracted from images. In addition, deep learning performs "end-to-end learning"– where a network is given raw data and a task to perform, such as classification, and it learns how to do this automatically as shown in figure 2.11.

In machine learning, you manually choose features and a classifier to sort images. With deep learning, feature extraction and modeling steps are automatic.

Another key difference is deep learning algorithms scale with data, whereas shallow learning converges. Shallow learning refers to machine learning methods that plateau at a certain level of performance when you add more examples and training data to the network [13].

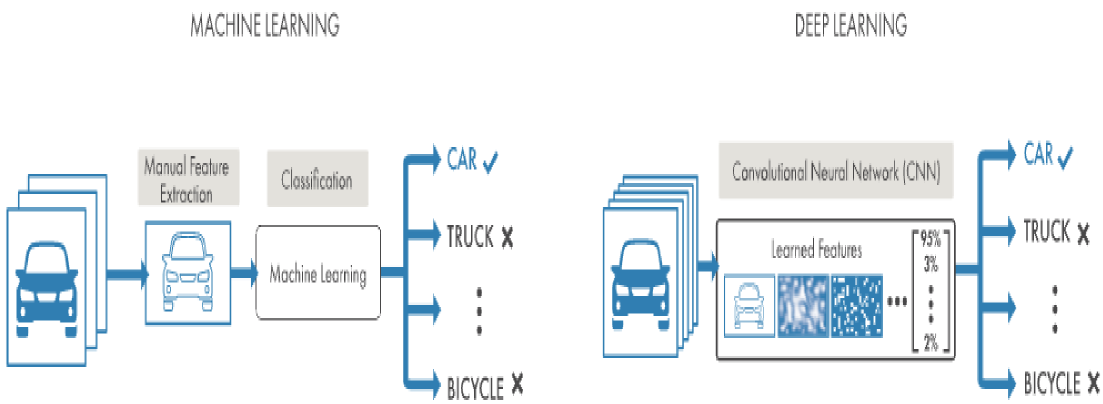

Figure 2.10: Difference Between Machine Learning and Deep Learning

2.3.5 Deep Learning Algorithms

- **Convolutional Neural Networks (CNNs)**

CNN's popularly known as Convent's majorly consists of several layers as shown in figure 2.12 and are specifically used for image processing and detection of objects. It was developed in 1998 by Yann Lacuna and was first called LeNet. Back then, it was developed to recognize digits and zip code characters. CNNs have wide usage in identifying the image of the satellites, medical image processing, series forecasting, and anomaly detection [6].

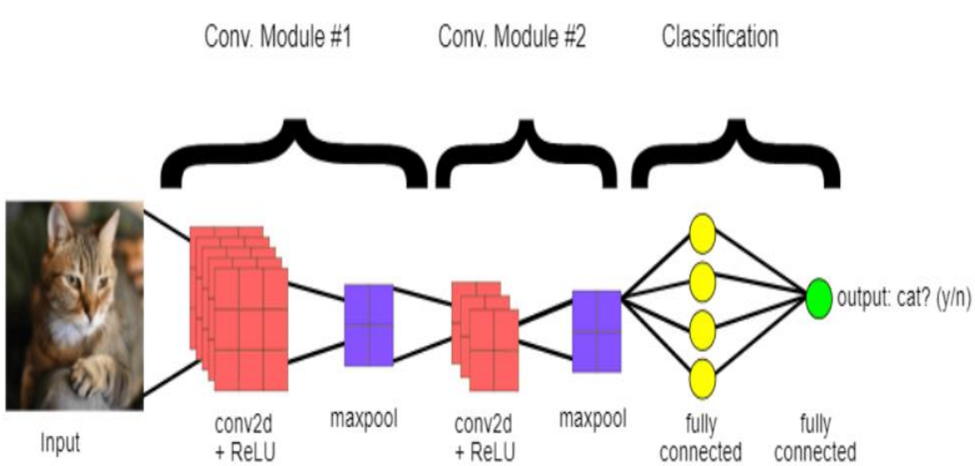

Figure 2.11: CNN layers

➢ Long Short Term Memory Networks (LSTMs)

LSTMs can be defined as Recurrent Neural Networks (RNN) that are programmed to learn and adapt for dependencies for the long term. It can memorize and recall past data for a greater period and by default, it is its sole behavior. LSTMs are designed to retain over time and henceforth they are majorly used in time series predictions because they can restrain memory or previous inputs. This analogy comes from their chain-like structure consisting of four interacting layers that communicate with each other differently. Besides applications of time series prediction, they can be used to construct speech recognizers, development in pharmaceuticals, and composition of music loops as well.

LSTM work in a sequence of events. First, they don't tend to remember irrelevant details attained in the previous state. Next, they update certain cell-state values selectively and finally generate certain parts of the cell-state as output. Below is the diagram of their operation [6].

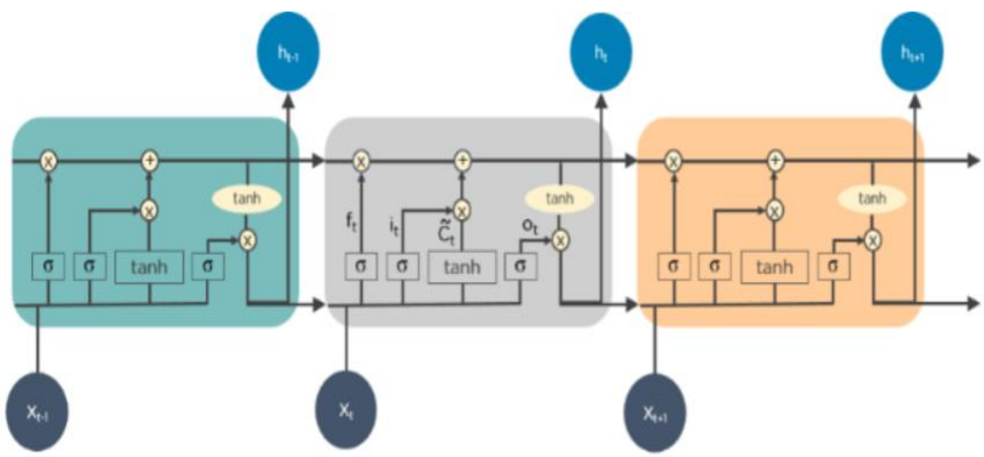

Figure 2.12: LSTMs steps

- **Recurrent Neural Networks (RNNs)**

  Recurrent Neural Networks or RNNs consist of some directed connections that form a cycle that allow the input provided from the LSTMs to be used as input in the current phase of RNNs. These inputs are deeply embedded as inputs and enforce the memorization ability of LSTMs lets these inputs get absorbed for a period in the internal memory. RNNs are therefore dependent on the inputs that are preserved by LSTMs and work under the synchronization phenomenon of LSTMs. RNNs are mostly used in captioning the image, time series analysis, recognizing handwritten data, and translating data to machines [6].

- **Multilayer Perceptrons (MLPs)**

  MLPs are the base of deep learning technology. It belongs to a class of feed-forward neural networks having various layers of perceptrons. These perceptrons have various activation functions in them. MLPs also have connected input and output layers and their number is the same. Also, there's a layer that remains hidden amidst these two layers as shown in figure 2.14, MLPs are mostly used to build image and speech recognition systems or some other types of the translation software.

  The working of MLPs starts by feeding the data in the input layer. The neurons present in the layer form a graph to establish a connection that passes in one direction. The weight of this input data is found to exist between the hidden layer and the input layer. MLPs use activation functions to determine which nodes are ready to fire. These activation functions include tanh function, sigmoid and ReLUs. MLPs are mainly used to train the models to understand what kind of co-relation the layers are serving to achieve the desired output from the given data set [6].

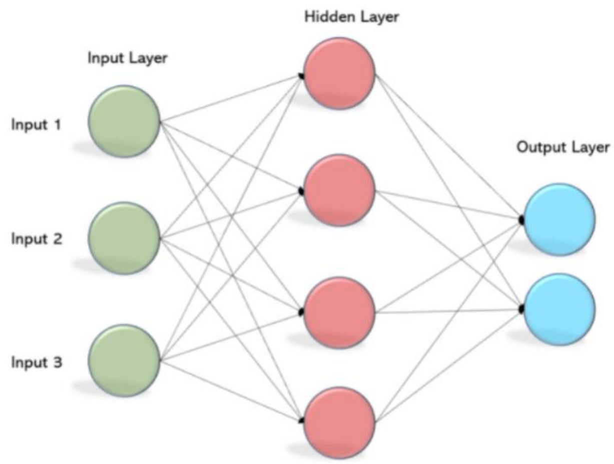

Figure 2.13: Multilayer Perceptrons

- **Generative Adversarial Networks (GANs)**

GANs are defined as deep learning algorithms that are used to generate new instances of data that match the training data. GAN usually consists of two components namely a **generator** that learns to generate false data and a **discriminator** that adapts itself by learning from this false data as shown in figure 2.15. Over some time, GANs have gained immense usage since they are frequently being used to clarify **astronomical images** and simulate **lensing** the gravitational dark matter. It is also used in **video games** to increase graphics for **2D** textures by recreating them in higher resolution like **4K**. They are also used in creating **realistic cartoons character** and also rendering human faces and **3D object rendering.**

GANs work in simulation by generating and understanding the fake data and the real data. During the training to understand these data, the generator produces different kinds of fake data where the discriminator quickly learns to adapt and respond to it as false data. GANs then send these recognized results for updating. Consider the below image to visualize the functioning [6].

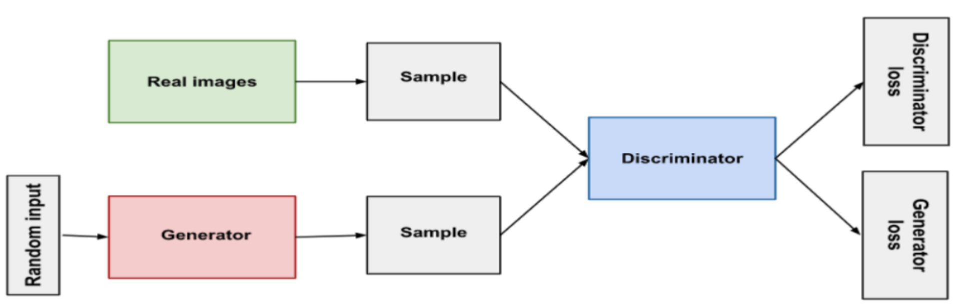

Figure 2.14:(GANs) steps

2.4 Convolution Neural Network (CNN):

2.4.1 Introduction Of CNN:

In the past few decades, Deep Learning has proved to be a very powerful tool because of its ability to handle large amounts of data. The interest to use hidden layers has surpassed traditional techniques, especially in pattern recognition. One of the most popular deep neural networks is Convolutional Neural Networks in deep learning.

CNN's were first developed and used around the 1980s. The most that a CNN could do at that time was recognize handwritten digits. It was mostly used in the postal sectors to read zip codes, pin codes, etc. The important thing to remember about any deep learning model is that it requires a large amount of data to train and also requires a lot of computing resources. This was a major drawback for CNNs at that period and hence CNNs were only limited to the postal sectors and it failed to enter the world of machine learning.

In 2012 Alex Krizhevsky realized that it was time to bring back the branch of deep learning that uses multi-layered neural networks. The availability of large sets of data, to be more specific ImageNet datasets with millions of labeled images and an abundance of computing resources enabled researchers to revive CNNs [22].

2.4.2 What Is CNN:

In deep learning, a convolutional neural network (CNN) is a type of artificial neural network commonly used for processing and analyzing visual imagery recognition. Compared with traditional neural networks, CNNs are more effective and efficient for image processing and natural language processing because they are specifically designed to analyze pixel data. Applications of CNNs include different computer vision tasks such as image and video recognition, image classification, image segmentation, and medical image analysis, as well as time series and brain-computer interfaces [23].

2.4.3 Applications of Convolutional Neural Networks:

Convolutional neural networks are used in a variety of different applications and are deployed quite frequently in many industries.

- **Facial Recognition**

Facial recognition technology relies on CNNs because the machines need to be able to detect changes in face shape over time to accurately identify people from one image to the next.

To do this, they must be trained using thousands of images containing faces from various angles and expressions. Once trained, they can compare new images with those stored in their database and determine whether they match or not [24].

- **Object Detection**

Object detection with a CNN works by training a model to recognize specific objects within digital images or videos by recognizing certain patterns, such as edges, shapes, and colors, that help distinguish one object from another.

The model is trained using labeled datasets—data points where each point has been assigned a label, such as safety vests or helmets. During training, the model learns how to recognize certain patterns associated with each label and maps them to corresponding labels when presented with new data points during inference [24].

- **Documentation Analysis**

CNNs offer numerous advantages over conventional rule-base systems when used for analyzing documents. For instance, they require much less effort than other techniques since limited human intervention is needed.

Secondly, since these are self-learning systems, they continue to get smarter over time, as they're capable of recognizing trends and patterns that humans might miss [24].

- **Biometric Authentication**

Biometric authentication technology, such as fingerprint scanners, has evolved considerably over the past decade. While there are several reasons why artificial intelligence can't replace humans at work just yet, technologies such as CNNs can definitely aid in making things easier.

When it comes to biometrics, CNNs can be used to identify very specific features in an individual's face or fingerprint that would be difficult or impossible for humans to detect manually**.**

For instance, if you want to authenticate someone using facial recognition technology, a CNN could scan through hundreds of images of that person's face and identify tiny details like pores or wrinkles that would be too small for humans to see with the naked eye [24].

- **Autonomous driving**

Images can be modelled using convolutional neural networks (CNN),which are used to model spatial information. CNNs are regarded as universal non-linear function approximators because of their superior ability to extract features from images such as obstacles and interpret street signs. Furthermore, as the depth of the network grows, CNNs may detect a variety of patterns. For instance, the network's initial layers will record edges, but its deeper layers will capture aspects like an object's shape that are more complicated (leaves in trees or tires on a vehicle). As a result, CNNs are the primary algorithm in self-driving cars [25].

2.4.4 Convolutional Neural Network Architectures:

A CNN architecture is divided into two components as shown in figure 2.15:

- ✓ In a process known as Feature Extraction, a convolution tool isolates and identifies the distinct characteristics of a picture for analysis. This feature extraction consists of an input, convolution layer, and pooling layer.
- ✓ Another component present in CNN architecture is classification in which we have fully connected the layer and output. The classification component is a fully connected layer that uses the output of the convolution process to forecast the image's class using the information acquired in earlier stages [26] [8].

A CNN typically has three layers: a convolutional layer, a pooling layer, and a fully connected layer as shown as figure 2.16.

Convolution Layer

The convolution layer is the core building block of the CNN. It carries the main portion of the network's computational load.

This layer performs a dot product between two matrices, where one matrix is the set of learnable parameters otherwise known as a kernel, and the other matrix is the restricted portion of the receptive field. The kernel is spatially smaller than an image but is more in-depth. This means that, if the image is composed of three (RGB) channels, the kernel height and width will be spatially small, but the depth extends up to all three channels.

During the forward pass, the kernel slides across the height and width of the image-producing the image representation of that receptive region. This produces a two-dimensional representation of the image known as an activation map that gives the response of the kernel at each spatial position of the image. The sliding size of the kernel is called a stride [27].

pooling layer

The pooling layer replaces the output of the network at certain locations by deriving a summary statistic of the nearby outputs. This helps in reducing the spatial size of the representation, which decreases the required amount of computation and weights. The pooling operation is processed on every slice of the representation individually.

There are several pooling functions such as the average of the rectangular neighborhood, L2 norm of the rectangular neighborhood, and a weighted average based on the distance from the central pixel. However, the most popular process is max pooling, which reports the maximum output from the neighborhood.

There are two main types of pooling:

Max pooling: As the filter moves across the input, it selects the pixel with the maximum value to send to the output array. As an aside, this approach tends to be used more often compared to average pooling.

Average pooling: As the filter moves across the input, it calculates the average value within the receptive field to send to the output array.

Fully Connected Layer

The name of the full-connected layer aptly describes itself. As mentioned earlier, the pixel values of the input image are not directly connected to the output layer in partially connected layers. However, in the fully-connected layer, each node in the output layer connects directly to a node in the previous layer**.**

This layer performs the task of classification based on the features extracted through the previous layers and their different filters. While convolutional and pooling layers tend to use Relu functions, FC layers usually leverage a SoftMax activation function to classify inputs appropriately, producing a probability from 0 to 1 [28].

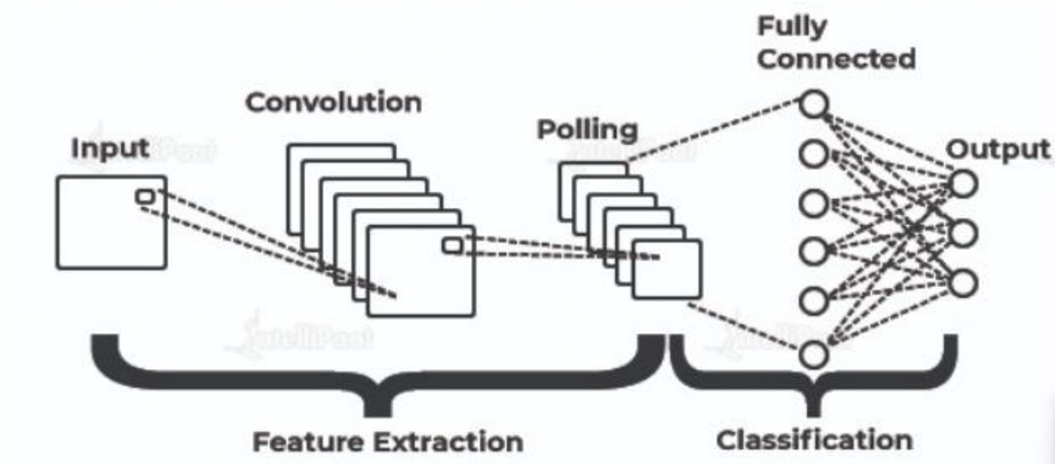

Figure 2.15:CNN Architecture

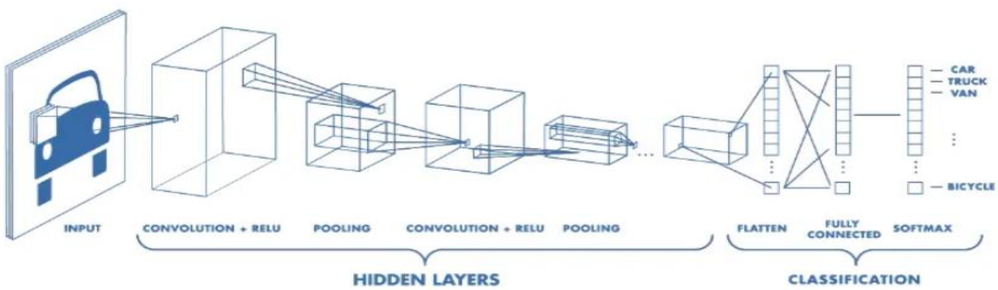

Figure 2.16:CNN layers

Non-Linearity Layers

Since convolution is a linear operation and images are far from linear, non-linearity layers are often placed directly after the convolutional layer to introduce non-linearity to the activation map.

There are several types of non-linear operations, the popular ones being:

a. **Sigmoid:**

The sigmoid non-linearity has the mathematical form $\sigma(\kappa) = 1/(1+e^{-}\kappa)$. It takes a real-valued number and "squashes" it into a range between 0 and 1.

However, a very undesirable property of sigmoid is that when the activation is at either tail, the gradient becomes almost zero. If the local gradient becomes very small, then in backpropagation it will effectively "kill" the gradient. Also, if the data coming into the neuron is always positive, then the output of sigmoid will be either all positives or all negatives, resulting in a zig-zag dynamic of gradient updates for weight.

b. **Tanh:**

Tanh squashes a real-valued number to the range [-1, 1]. Like sigmoid, the activation saturates, but unlike the sigmoid neurons ,its output is zero centered.

c. **Relu:**

The Rectified Linear Unit (Relu) has become very popular in the last few years. It computes the function $f(\kappa)=\max(0,\kappa)$. In other words, the activation is simply threshold at zero.

In comparison to sigmoid and tanh, Relu is more reliable and accelerates the convergence by six times.

Unfortunately, a con is that Relu can be fragile during training. A large gradient flowing through it can update it in such a way that the neuron will never get further updated. However, we can work with this by setting a proper learning rate [27].

2.5 deepfake:

2.5.1 what is deepfake?

Deepfakes are artificial media produced using deep learning techniques and a portmanteau of “deep learning” and “fake.” Deepfakes replace features on one image with those of another [29].

The term "deepfake" comes from the underlying technology "deep learning," which is a form of AI. Deep learning algorithms, which teach themselves how to solve problems when given large sets of data, are used to swap faces in video and digital content to make realistic-looking fake media [30].

2.5.2 How do deepfakes work?

Deepfake videos commonly swap faces or manipulate facial expressions as shown in figure 2.17. The image below illustrates how this is done. In face swapping, the face on the left is placed on another person’s body. In facial manipulation, the expressions of the face on the left are imitated by the face on the right.

Deepfakes rely on artificial neural networks, which are computer systems that recognize patterns in data. Developing a deepfake photo or video typically involves feeding hundreds or thousands of images into the artificial neural network, “training” it to identify and reconstruct patterns—usually faces [31].

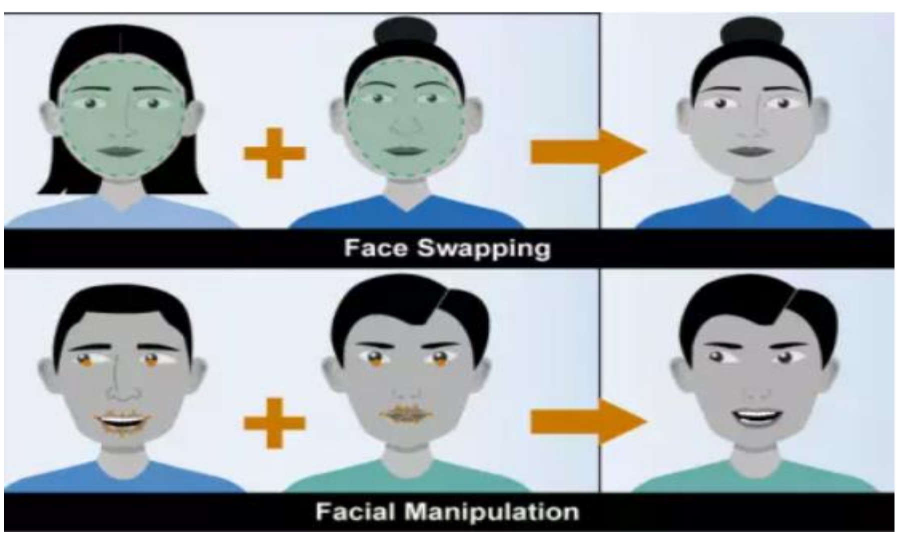

Figure 2.17:deepfake

2.5.3 methods of deepfake:

- **Autoencoders:**

  The autoencoder is a deep learning AI program tasked with studying the video clips to understand what the person looks like from a variety of angles and environmental conditions, and then mapping that person onto the individual in the target video by finding common features [30].

  The first step to producing a deepfake is transforming the face images into smaller feature-based representations using an encoder. This more information-rich representation is often referred to as the latent face. The latent face will contain representations for features such as the nose shape, skin tone and eye color. We use the same encoder for each person, so the representations produced have the same meaning.

  We then transform the latent face back into an image using a decoder. Depending on which images we use to train the decoder, the output face image will vary. The key part for face swapping is that the decoder for person A is applied on the latent face of person B, and vice versa. In this way, the output face will have the expression and structure of person A, but the style and look of person B [29].

- **Generative Adversarial Networks (GANs):**

  GANs are also used as a popular method for creation of deepfakes, relying on the study of large amounts of data to "learn" how to develop new examples that mimic the real thing, with painfully accurate results [30].

  GAN uses deep learning to recognize patterns in real images and then uses those patterns to create the fakes. When creating a deepfake photograph, a GAN system views photographs of the target from an array of angles to capture all the details and perspectives. When creating a deepfake video, the GAN views the video from various angles and analyzes behavior, movement and speech patterns. This information is then run through the discriminator multiple times to fine tune the realism of the final image or video [32].

GANs have a smarter system, which includes a generator and a discriminator as shown in figure 2.18. The former reproduces data it's learned into deepfakes that must then fool the latter.

The discriminator compares the generator's creations against real images and determines their effectiveness [33].

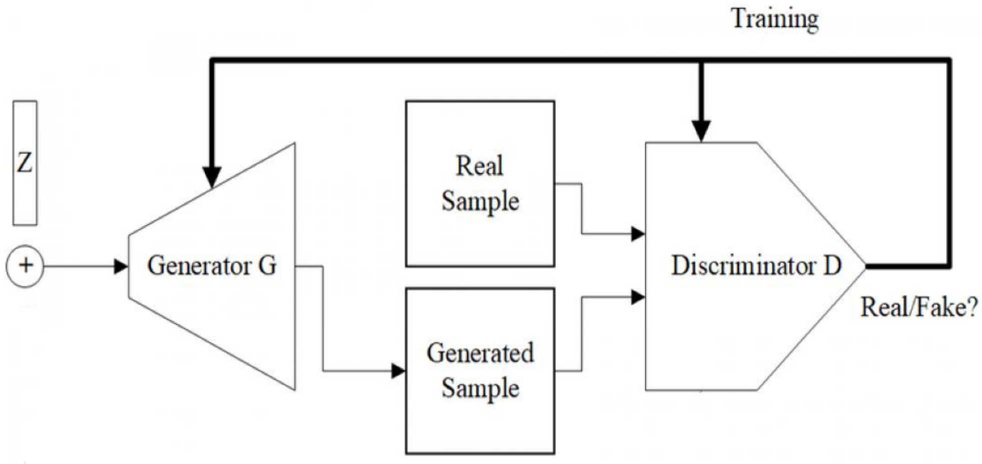

Figure 2.18:Generative Adversarial Networks

2.5.4 Positive Applications of Deepfake Technology:

- **Education**

Deepfakes can help teachers deliver engaging lessons. Additionally, these courses will go beyond traditional visual and media formats.

AI-generated synthetic media can bring historical figures to life in the classroom. This makes the lessons more engaging and interactive. Synthetic video reenacted or voice and video of historical figures will have an even greater impact. It can increase engagement and be a more effective learning tool.

- **Accessibility**

AI can create tools that can hear, see, and reason with increasing accuracy very quickly. Thanks to Artificial General Intelligence (AGI). AI-generated synthetic media can also help people amplify their agency. It gives accessibility tools independence by making them smarter, more affordable, and more customizable. Additionally, AI-based tools can make solutions more accessible to everyone.

- **Innovation**

  In many industries, data and AI are assisting with digital transformation and automation. Deepfake is also gaining traction as a way to engage customers and deliver value.

  The deepfake approach allows brands to create a virtual trial room. Here users can try out products before purchasing them. Retailers can also engage customers at home by creating a mixed reality world powered by AI. It would allow them to try on furniture and decorate their space.

  Low-resolution images can also use AI to enhance and improve their resolution. These deepfake enhancement techniques are especially useful for older media [34].

2.5.5 Dangerous Applications of Deepfake Technology:

- **False Information/Fake News**

  Fake news isn't new, and its use is to sow discord and division throughout history. It is still used today to deceive the public and disrupt political, business, and social activities. Fake videos that depict real events or show real people saying and doing things they never did can sow doubt and confusion. The fake news industry is already attempting to do so.

- **Fake Videos**

  A video showing a Thai actor replacing President Trump was created using the first version of deepfake technology. The distorted video was widely shared on social media and received widespread attention. Other videos have gone viral, with the actress in the original video claiming the president is real, but the president claiming he is not.

- **Extorting Money from Businesses**

  Manipulated with deepfake, faces and voices transferred to media files show people making false statements. It is possible to make a video of a CEO making fake announcements. An attacker could also blackmail a company by threatening to send the video to press agencies or post it on social media [34].

2.5.6 First Order Motion Model for Image Animation

Let's look at how this method works. The whole process is divided into two parts: **motion extraction** and **motion generation**. The source image and the driving video serve as input. The motion extractor uses an automatic encoder to detect key points and extracts a first-order motion representation composed of scattered key points and local affine transformations. These are used in conjunction with the guidance video to generate a dense optical flow and occlusion map with a dense motion network. The moving dense array and the source image outputs are then used by the generator to render the target image as described in figure 2.20.

This work outperforms state of the art on all the benchmarks. Apart from that it has features that other models just don’t have. The really cool thing is that it works on different categories of images, meaning you can apply it to face, body, cartoon, etc. This opens up a lot of possibilities. Another revolutionary thing with this approach is that now you can create good quality Deepfakes with a single image of the target object [35] [6].

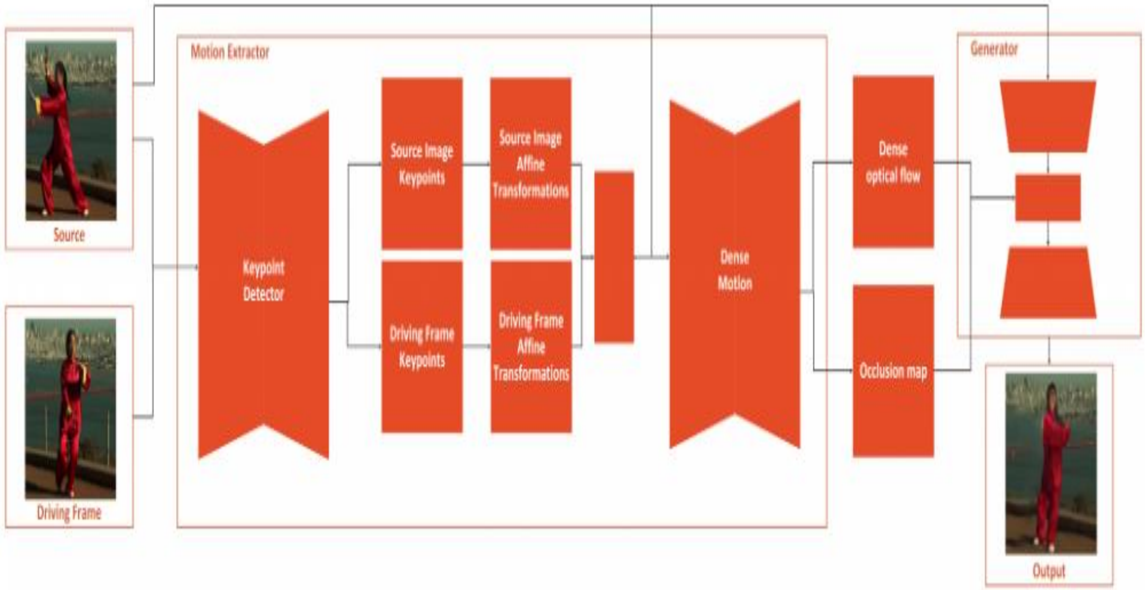

Figure 2.20:First Order Motion

Figure 2.19:learned key points in first order motion.

3) Chapter 3: System Architecture and Methods

3.1 System Architecture

System architecture plays an important role in defining the structure and behavior of our project. It helps us to better understand how the project works and ensures that it functions as intended throughout its lifecycle. This includes defining the relationships between components, data flows, service compositions, and subsystems. by designing an effective system architecture that prioritizes scalability, and interoperability, among other objectives, and helps us to have greater confidence in the system's ability to meet its requirements and objectives [36].

System architecture has five layers such as Application layer, Image recognition layer, voice recognition layer, deepfake layer and database layer as shown figure 3.1, we explain more details about each layer in the following section.

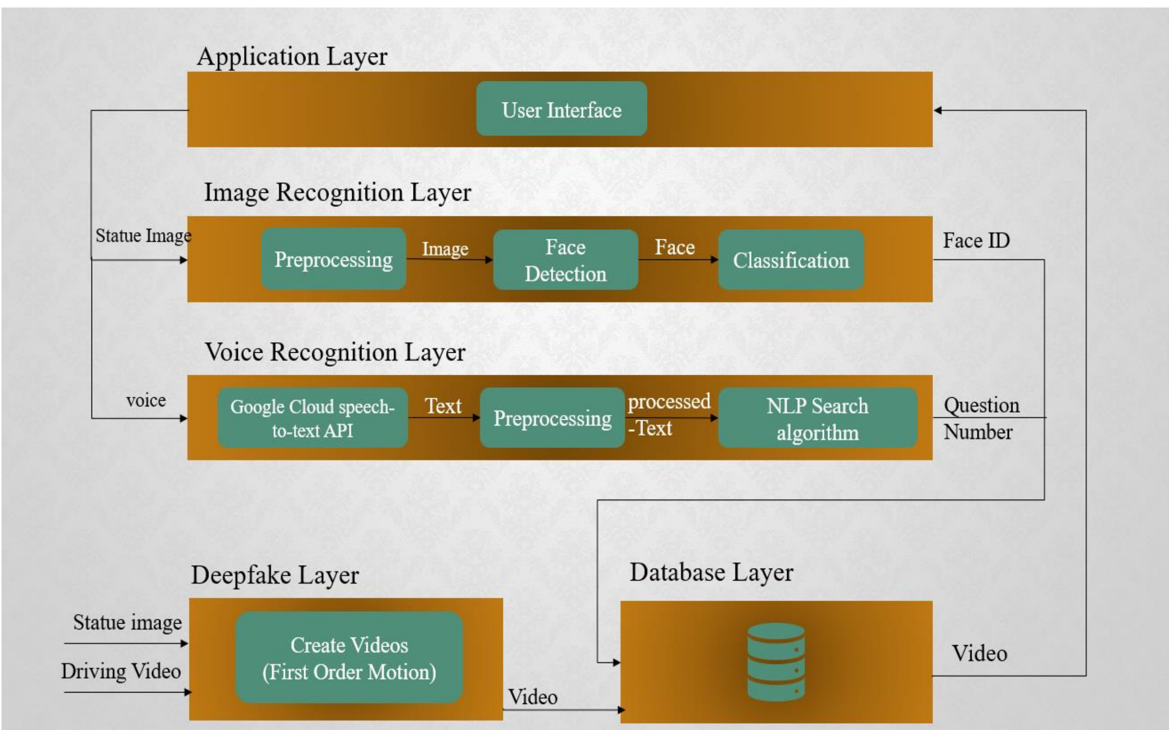

Figure 3.1:System Architecture

3.2Description of methods and procedures used

Application Layer:

The Android application enables users to access a wealth of information about statues by simply taking a photo of them using their mobile phone camera or selecting an image from their phone's image gallery. Once a statue has been recognized by the application, users are able to engage in a virtual conversation with the statue. The user is prompted to ask a question, to which the application responds with a fake video of the statue providing an answer. This feature provides an immersive experience for users, allowing them to engage with historical and cultural artifacts in a new and exciting way. By leveraging advanced image recognition and video generation technologies, this application represents an innovative approach to interacting with art and history on mobile devices.

Face Recognition Layer:

We used in this layer the CNN model to identify and classify the image of the statue that the user entered into the application through the phone's camera or a photo in the phone's gallery.

There are many architectural models of deep convolutional neural networks (CNN) such as the VGG16, VGG19, ResNet (residential Network), Inception, GoogLeNet, AlexNet and Custom CNN [37] [38].

We tried them all and concluded that the best architectural model in terms of results, speed, and accuracy is VGG-19.

CNN model with VGG-19 architecture:

VGG-19 is a deep neural network with 19 layers, and it consists of 16 convolutional layers and 3 fully connected layers as described in figure 3.2.

The convolutional layers are grouped into five blocks, with each block consisting of multiple layers with the same kernel size of 3x3. The number of filters in each block increases as we move deeper into the network, with the first block having 64 filters and the last block having 512 filters as shown in figure 3.2 [39] [40].

The pooling layers used in the VGG-19 architecture are max pooling layers with a pool size of 2x2 and stride of 2 as shown in figure 3.2 [39] [40].

The fully connected layers at the end of the network have 4096 neurons each, followed by a final output layer with several neurons equal to the number of classes in the classification task as shown in figure 3.2 [39] [40].

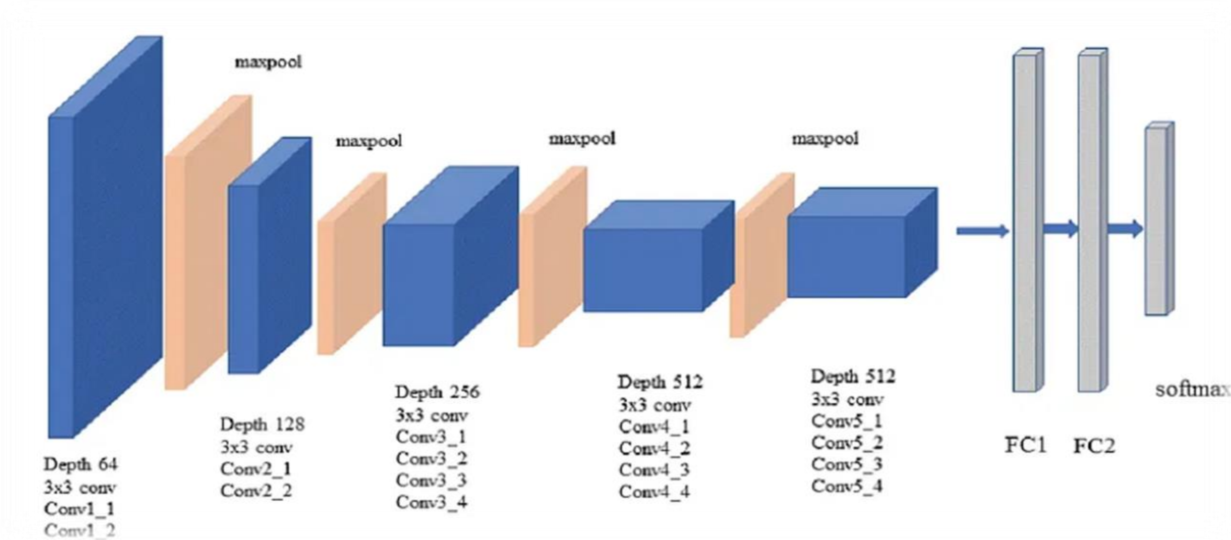

Figure 3.2:VGG19 Architecture

CNN Parameters:

- **Activation Function**: an activation function is a mathematical function that is applied to the output of a neuron in a neural network. It introduces non-linearity into the output of the neuron, which is important for the network to be able to learn complex patterns and relationships in the input data [41].

  We tried many Activation functions with our architecture such as Tanh, Sigmoid and Relu in the fully connected layers but **Relu** got the best accuracy with our model and used **Softmax** activation function in the last fully connected layer to normalize the output of each class with a probability from range (0 to 1).

- **Number of epochs:** we tried varies number of epochs with our model and finally the best range of epochs between 15 and 20.

- **Batch size:** it is a hyperparameter that specifies the number of training examples used in each iteration of the training algorithm during the training process. In other words, the batch size determines how many training examples are processed at once before the weights of the neural network are updated [42].

  A small batch size can result in more frequent updates to the weights of the network, which can improve the speed of convergence and lead to faster training. However, using a small batch size can slow down the convergence of the network.

  On the other hand, using a large batch size can result in more stable weight updates and faster convergence, but it can also require more memory and computational resources.

  We tried many numbers of batch size and the best number with our model is **16.**

Loss Function:

Loss Function is a fundamental component of machine learning algorithms that helps to optimize model parameters, guide model selection, handle imbalanced datasets, and regularize the model. Choosing an appropriate loss function is an important step in designing a machine learning model for a specific task [43].

We used **categorical cross entropy** loss function with our architecture.

Optimization of CNN model:

- We used different optimizers such as Adam, RMSprop, and Momentum SGD, we made many experiments with these optimizers on our architecture used with different hyperparameters and the optimizers that leaded us to the best results was **Adam** with **learning rate is 0.001.**

- We used **Early Stopping** to monitor the validation loss and prevent the overfitting that occurred during the training process as shown in figure 3.3.

```
History = full_model.fit(X_train, y_train, epochs=20, batch_size=16,validation_data=(X_test, y_test),workers=10,callbacks=[EarlyStopping(monitor='val_loss',
                                    patience=4,mode = 'auto',
                                    restore_best_weights=True)])
```

Figure 3.3:fit model with early stopping

- We used **ImageNet** weight to our architecture to improve the model performance by using the previously trained model on a large data set (such as ImageNet) and transferring knowledge and features to VGG-19 and it helps the model converge in less epochs.

  ImageNet is an Image database consisting of 14,197,122 images organized according to the WordNet hierarchy, ImageNet is formally a project aimed at (manually) labeling and categorizing images into almost 22,000 separate object categories for the purpose of computer vision research as shown in figure 3.4 However, when we hear the term "ImageNet" in the context of deep learning and Convolutional Neural Networks, we are likely referring to the ImageNet Large Scale Visual Recognition Challenge, or ILSVRC for short, The goal of ImageNet image classification challenge is to train a model that can correctly classify an input image into 1,000 separate object categories, Models are trained on ~1.2 million training images with another 50,000 images for validation and 100,000 images for testing, These 1,000 image categories represent object classes that we encounter in our day-to-day lives, such as species of dogs, cats, various household objects, vehicle types, and much more [44] [45] [46].

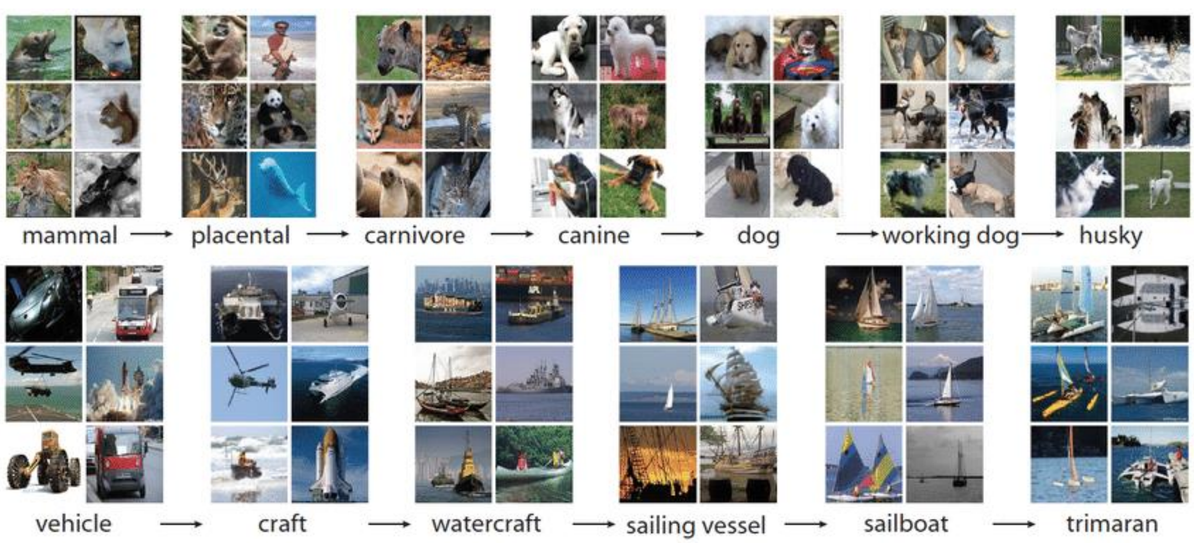

Figure 3.4:ImageNet Dataset

Deep Fake Layer:

We use deep fake models in our project to create a fake video of the statue in which he speaks and gives information about himself.

We tried many deep fake models such as First Order Motion Model, Make It Talk Model and Face2Face and we found that the best model to create fake videos of statues in our project is First Order Motion Model according to results, speed, and accuracy.

First Order Motion Model implementation:

First order motion needs Two main inputs:

- Source Image: image of statue as shown in figure 3.6.
- Driving video: video of a person speaking as the statue and describing himself as shown in figure 3.5.

Figure 3.6:source Image

Figure 3.5:Driving video.

First order motion consists of Two main modules as described in figure 3.7:

- Motion Extraction Module.
- Image Generation Module.

Motion Extraction Module consists of an encoder that learns a latent representation containing sparse key-points of high importance in relation to the motion of the object.

In Motion Extraction Module as in shown in figure 3.7:

- **Sparse key-points:** The movement of these key points across the different frames of the driving video generates a motion field.
- **Dense motion:** predicts the motion of every individual pixel of the frame.
- **Occlusion Map:** highlights the pixels of the frame that need to be in-painted.

Image Generation Module (Appearance Module) uses an encoder to encode the source image, which is then combined with the Motion Field and the Occlusion Map to animate the source image.

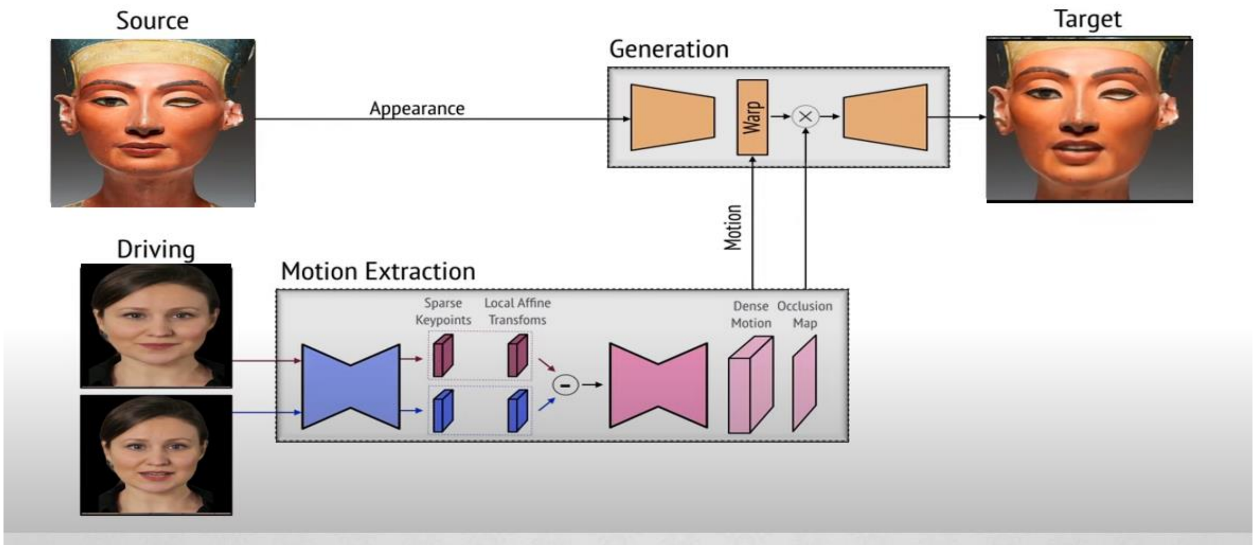

Figure 3.7:First Order Motion Model

First Order Motion preprocessing:

The following preprocessing refers to the steps that are taken to prepare the input images and the driving video before they are fed into the model for animation.

The preprocessing steps typically involve the following:

- **Resizing**

Resizing is an important step in the First Order Motion Model, as it ensures that the input data has a consistent size and shape, which is necessary for the model to learn and apply the motion and appearance of the driving video to the source image [6].

Resizing involves changing the dimensions of the source image and the driving video frames to a fixed size, such as 256x256 pixels. This can be done using image processing libraries such as OpenCV or Pillow. Resizing the images to a smaller size can also help reduce the computational load on the model during training and inference.

- **Cropping**

Cropping is an important step in the First Order Motion Model, as it helps to focus the model's attention on the main subject of the animation and improve the overall quality of the output. By selecting a smaller region of the input data, the model can more easily learn and apply the motion and appearance of the driving video to the source image, resulting in more realistic and accurate animations [6] [47].

Cropping is an important step in the First Order Motion Model, as it helps to focus the model's attention on the main subject of the animation and improve the overall quality of the output. By selecting a smaller region of the input data, the model can more easily learn and apply the motion and appearance of the driving video to the source image, resulting in more realistic and accurate animations[44].

- **Normalization**

Normalization is an important step in the First Order Motion Model, as it helps to ensure that the input data has a consistent scale and avoids any potential numerical issues during training and inference. By scaling the pixel values of the input data to a standard range, the model can more easily learn and apply the motion and appearance of the driving video to the source image, resulting in more accurate and realistic animations [6].

Normalization involves scaling the pixel values of the input data so that they have a mean of zero and a standard deviation of one. This can be done using a variety of normalization techniques, such as min-max scaling or Z-score normalization [6].

- **Grayscale Conversion**

Grayscale conversion is an optional step in the First Order Motion Model, but it can be useful for reducing the computational load on the model during training and inference. Grayscale images have a single channel of data, compared to the three channels (Red, Green, Blue) in color images, which means that they require less memory and computation to process. However, grayscale conversion can also result in a loss of information, particularly in applications where color is important, such as image and video editing [6].

Grayscale conversion involves transforming the RGB color values of each pixel in the image to a single grayscale value, which represents the brightness of the pixel. This can be done using a variety of techniques, such as taking the average of the RGB values or using a weighted average based on the luminance of each color channel.

Voice Recognition Layer:

Users can interact with a virtual statue to ask questions and receive fake videos containing answers. The process involves the user choosing a random statue, which then leads to the Conversation page. In the Conversation page, the user can either record a question or choose a question from a list of predefined questions. Once the user submits the question, the statue analyzes the question and returns a fake video containing the answer.

Steps of Voice Analysis Algorithm as shown in figure 3.8:

- The Application gets record from user.
- Send the record to Google Cloud API to get text.
- Apply the preprocessing on the text.
- Use NLP Linear Search Algorithm to get Question number.

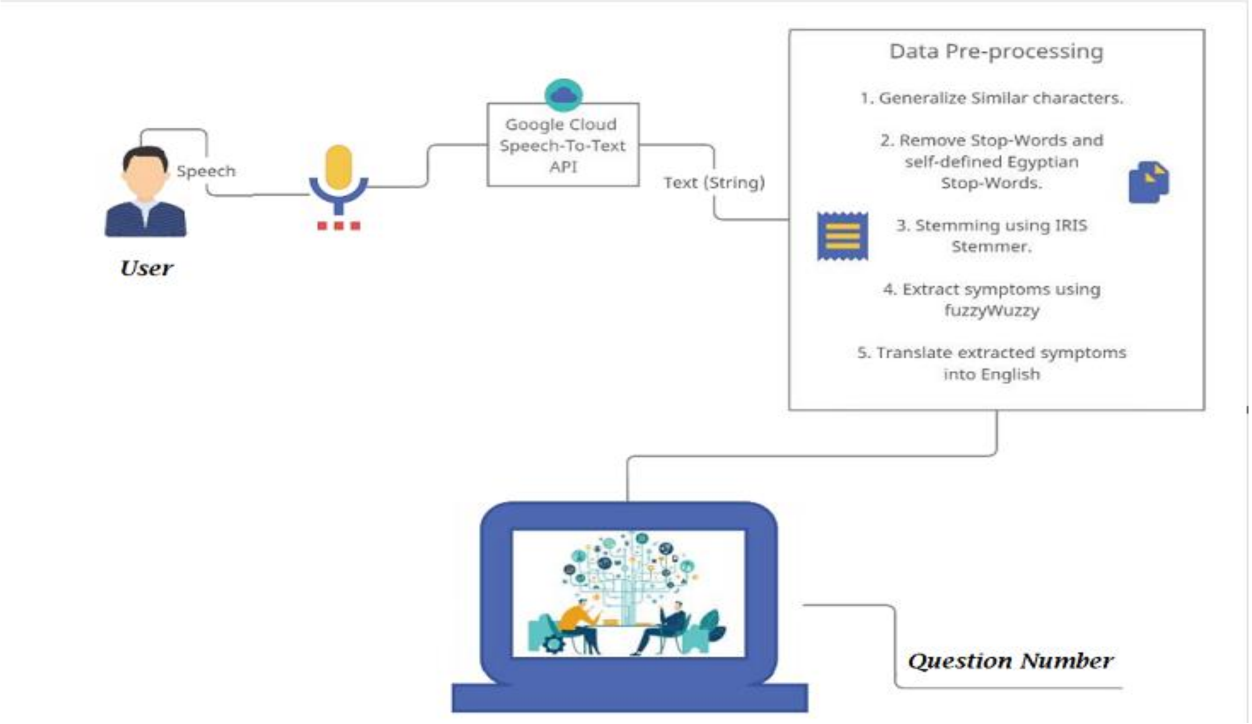

Figure 3.8: Voice Analysis Algorithm

Preprocessing on the text:

- **Generalizing similar characters:**

  Generalizing similar characters is an important step in NLP preprocessing, as it can help improve the efficiency and accuracy of text processing and analysis by reducing the number of distinct characters that need to be processed [48].

  Generalizing similar characters refers to a step where similar characters are grouped together or replaced with a common symbol or character. This is done to simplify the text data and reduce the number of distinct characters that need to be processed [48].

  For example, in English text, the characters "l" (lowercase L) and "I" (uppercase i) can be difficult to distinguish from each other due to their similar appearance. By generalizing these characters, they can be replaced with a common symbol, such as "|" or "/", making it easier to process and analyze the text [48].

- **Removing Stop Words:**

  stop words are common words that are removed from the text data to simplify the data and improve the efficiency and accuracy of text processing and analysis [48] [49].

  Removing stop words can help simplify the text data and reduce the noise in the data, making it easier to process and analyze. However, it is important to note that removing stop words can also result in a loss of context and meaning, particularly in applications where the stop words are important for understanding the text, such as sentiment analysis or topic modeling [49].

  Stop words are words that occur frequently in the language and do not carry much meaning or information, such as "the", "to", "and", "of", "in", "a", and so on [49].

- **Stemming:**

  Stemming can help improve the efficiency and accuracy of text processing and analysis by reducing the number of distinct words that need to be processed and by normalizing the words to their base form. However, it is important to note that stemming can also result in a loss of information and context, particularly in languages with complex morphology or words with multiple meanings [48] [50].

  There are various stemming algorithms and libraries that can be used for stemming in NLP preprocessing, including the IRIS stemmer [50].

  For example, the stem of the words "running", "runner", and "runners" is "run".

Database Layer:

The Database Layer acts as a central repository for storing and managing system data. It provides the infrastructure for administrators and database managers to control and manipulate system resources.

This layer is responsible for storing the generated videos, user queries, and other relevant data, ensuring efficient retrieval and management.

The Database Layer supports functionalities such as video creation, editing, deletion, and addition, facilitating seamless integration with the other layers of the system.

4) Chapter 4: System Implementation and Results

4.1 Dataset

4.1.1 Dataset introduction

A dataset is a set of structured and interconnected data, is considered an essential component for the development of machine learning and data learning models and is considered a powerful tool for understanding data and using it to achieve various goals in various fields and applications.

Documentation for a dataset should include information about the data sources, the preprocessing techniques used, and the format of the dataset. It should also include any specific instructions for loading and using the dataset in a machine learning framework. Additionally, it is helpful to include information about any potential biases in the data and any limitations or caveats to be aware of when using the dataset.

Overall, creating a high-quality dataset is a critical step in building effective machine learning models. By following best practices for data collection, preprocessing, and organization, you can ensure that your dataset is clean, consistent, and relevant to your problem, making it easier to train and test your models effectively.

4.1.2 Dataset challenge

In the beginning, we spent a lot of time searching for a ready-made dataset for our project but we had difficulty finding information and images for each statue, so we decided to collect the dataset ourselves, and it was difficult to get a large number of images of statues in high resolution from the internet, so we decided to head over to Egyptian museums to collect many images of statues with special standards to increase the accuracy and quality of the project.

4.1.3 Dataset preparing and content.

The data set is mainly divided into three parts:

- **Text**
  The textual dataset that contains all the information about the statues of kings and queens collected, verified and reviewed from documented historical sources and references such as "Egyptian Civilization" by John Romer and articles published in well-known historical journals such as "Egyptian Journal of Archeology" and "Egyptian Museum Bulletin" And the "Journal of Ancient Egypt" [6-7] and some professors from the Faculty of Archeology.

Sample of textual dataset as shown in figure 4.1.

	personal Info	Statues and information about them	The full family / reign period	his works	The current location of the statue (statues)	Tomb or Statue Finder (Names / Dates)	Oddities and wonders happened to the st
Senusret I	I am King Senusret I, and my name means a man of strength, also pronounced in English as Sesostris I. I am the son of King Amenemhat I and Queen Nefertiten, and my wife is Neferu the third, as was my sister too. My sons are Amenemhat II and the princesses Itakait	They found 10 statues of me and the statues were made of limestone. This group depicts me in my youth when I was sitting on the throne.	I was a king from the middle Egyptian state and I am the second king of the twelfth dynasty. I ruled Egypt for 45 years from 1971 to 1926 BC.	I continued my father's expansionist policies of an offensive nature in Nubia, by initiating two expeditions to this region, and adopted the official southern borders of Egypt near the second waterfall, where a defensive garrison was placed and a victory plaque was erected on it. It also organized a campaign for the oases of the Western Desert. And established	The statue is now in the Egyptian Museum	My group of statues was discovered in 1894 inside a pit in the funerary temple of my pyramidal group in the Lisht area of Fayoum	
Khafre	Many people call me Khafre, but my name is Khaf-Ra, which means (transfiguration like Ra). They called me Sekhem Eb, which means strong in heart, and Hoor Nefer, which means good Horus. My father is King Khufu. I married 4 wives, including Princess Meras Ankh, and	My statue was found in the Valley Temple of my hierarchical group for the king, its height is 168 cm. The statue is characterized as two sides of the same coin. The king is the god Horus on earth and the god Horus is in the sky. This statue confirms the association of the Trinity, where the chair represents Isis, the king	I was in the era of the Old Kingdom and my order was the fourth king of the Fourth Dynasty, and I ruled for 24 years from 2559 to 2535 BC.	My most important work is the creation of my hierarchical group in Giza, which is characterized by the presence of the Valley Temple in which the most beautiful statues of the ancient Egyptian civilization were found. The statue of the Sphinx is attributed to me, as researchers called it (Khar - Um - Akhi) meaning Horus on the horizon, and I have inscriptions in	The statue is now in the Egyptian Museum	The statue was found in the Valley Temple of the King by the excavations team of Harvard University and the Boston Museum under the leadership of archaeologist George Reisner in 1910.	
Senusret III	I am King Senusret III, my original name is (Sen Osert) and my royal name is (Khacka Ra), which means the power of Ra is coming. My father is Senusret II, my mother is Nefert Hajit the first, my wives are Marit Seger, Nefert Hanut Khanmit,	Statue of Senusret III, made of gray granite. Its height is 122 cm. The features of the statue follow the realistic school, and technically, the statue is reproached with an excessive elongation in the arms	I was in the era of the Middle Kingdom and the 12th dynasty, and I was the fifth king of the family. And I ruled for 38 years from 1878-1839 BC	They consider me the greatest of the 12th dynasty kings. I built the first water canal linking the Mediterranean and the Red Sea through the Nile River. This canal was called the Sesostris Canal. And he established its extension and called it a canal (Kabrit).	The statue is now in the Egyptian Museum	The statue was discovered by the Egyptian-German mission working in the Matariya archaeological area in 2005	
ar "Sheikh Al Bal	I am a statue of Kabar, also known as Sheikh Al Balad. Also known as the High Priest	The statue was found in the Kabar Mastaba in Saqqara, north of the pyramid of Djoser. The statue is 112 cm high and has other statues accompanied by his wife. The statue was made of wood (sycamore) and the eye	I lived in the era of the Old Kingdom, between the end of the Fourth Egyptian Dynasty and the beginning of the Fifth Egyptian Dynasty from 2465 to 2323 BC	I was the governor of a province and then a clerk to the king in the army and I was associated with some military campaigns in the south of the Levant.	The statue is now in the Egyptian Museum	My statue was discovered by Auguste Mariette in the Saqqara necropolis during excavation in 1860.	We find the eye in the statue inlaid with rock crystal, showing the s alive, and it is an application of the thought and belief that domina Egyptian thinking in terms of immortality and rebirth.
Nefertiti	I am Queen Nefertiti, whose name means "The Beautiful Has Come". I am the wife of King Amenhotep IV (who later became Akhenaten), ruler of the Eighteenth Dynasty. She was one of the most powerful women in ancient Egypt. I lived for a short period after my husband's death and helped Tutankhamun to take over the kingdom. I had a high status during her husband's rule.	I am famous for the bust of my face depicted, carved on a piece of limestone. One of the most wonderful pieces of art from the ancient era, it is the most famous drawing of the queen, and one of the most famous pieces of sculpture ever. Made of colored limestone and some gypsum additions, it is about 50 cm high. This piece is distinguished as a rare piece in which the artist succeeded in accurately expressing the anatomical lines of the neck and the structure of the facial bones. There is another statue made of brown quartz and its height is 32.5 cm. This piece, next to the Berlin Museum piece, can be considered one of the best and best examples we know of Queen Nefertiti	I belong to the eighteenth family, and I lived in the fourteenth century BC from 1370 to 1330 BC.	Queen Nefertiti shared her husband in the worship of the new god Aten, the power of the sun disk, and she and her husband were the mediator between the people and Aten, and it is assumed that the full blessing be granted only when the royal couple united. During the early years of her husband's rule, Nefertiti changed her name according to the change of her creed to Neferneferaton Nefertiti, which means "the furnace that rises because the beautiful has come."	There are two pieces, one currently in the Berlin Museum and the other in the Egyptian Museum	The German Egyptologist Ludvig Borchardt found the famous statue on December 6, 1912 in the workshop of Tuthmosis the sculptor in Amarna.	
Khufu	I am King Khufu and my personal name is (Khnum Khufu) meaning (Khnum protects me) and the Hurrian name is (Majd-R-Nabati). My father, King Sneferu, and my mother, Queen Hotep heres I. My wives are (the first Meritites, Hanutsen) and my sons are Kawab, Dedaf Hor, Hetepe Haras the second, the second Meretitis, the second Mers Ankh, Bafara, Dedaf Ra, Min Khaef I, Khafre, Khufu Khaef I, Bail I, Hor Baeef, Nefert Yapet.	My statue found in the temple of the god Khenti Amentio in Abydos is made of ivory, and its height is about 7.5 cm.	I ruled during the Fourth Dynasty of the Old Kingdom, about 2580 BC, where I was the second ruler of the Fourth Dynasty. Estimates vary as to the exact period of his reign. Where the Turin papyrus, which is from the era of the modern state and is considered one of the most important documents for the succession of the kings of Egypt, mentions that Khufu ruled for 23 years. The Greek historian Herodotus says that he ruled for 50 years, and the Egyptian priest Manetho, who lived in the third century BC, says that he ruled	I built the Great Pyramid of Giza, the highest pyramid in the world. The pyramid was named "Akhet Khufu", meaning "the horizon of Khufu". The Great Pyramid is one of the Seven Wonders of the Ancient World, and is the only one that remains.	The statue is now in the Egyptian Museum	A nthy scientist found the statue in 1903 AD in Abydos, southern Egypt.	
Akhenaten	Aten." Before the fifth year of my reign, I was also known as Amenhotep IV, meaning (Amon pleases). It was among the titles I bore (Ankh um Maat). I was the second son of King Amenhotep III and became the crown prince after the death of the eldest son, Prince Tuthmosis. My wives are Queen Nefertiti, who is the most famous, Kia and Ankhesen Amun, and her name means (she lives through Amun). I had 6 daughters from my main wife, Queen Nefertiti, and my	I have a limestone statue, 40 cm high, as it represents me as I present the offering table. I also have a group of statues, each about 4-7 meters high, made of sandstone. These statues reflect my familiar appearance in an exaggerated way.	I was a ruler of the 18th Dynasty and ruled Egypt for about 17 years from 1353 to 1336 BC. My main wife, Nefertiti, participated with me in the ruling.	I was the owner of the first religious revolution in history. As the first years of my rule passed, I shared power with my father in the capital, Thebes. But I was determined to achieve a radical religious reform, so I left the capital, Thebes, with the consent of my father the king, and to the north of Thebes, "away from the land of Amun," a new city soon became the thriving capital of Akhet Aten (Tel el-Amarna).	The statue is now in the Egyptian Museum	Henri Chevrier revealed a group of Osiris statues of King Akhenaten in the open courtyard of the temple of Aten in Karnak. They are about the remains of 24 square column bases erected in front of each of them is a huge statue of King Akhenaten, each about 4-7 meters high, made of sandstone. A colorful limestone	

Figure 4.1:textual dataset

- **Videos**

  The dataset of videos created by the admin in the desktop application which uses Deepfake model, and each video contains historical information provided by the statue itself as shown in figure 4.3, and the Firebase platform is used to store the videos.

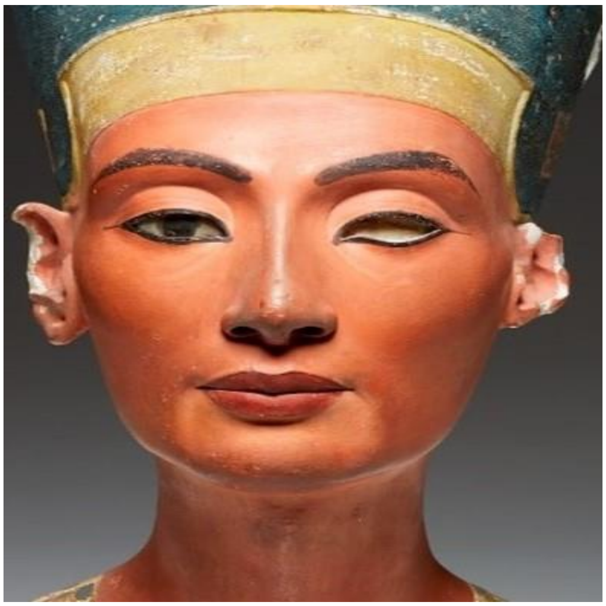

Figure 4.2:before generation

Figure 4.3:After generation

- **Images**

The Images of Statues**,** We faced a big problem, which is the lack of high-quality images and the small number of images that the model needs, available for free on the Internet, as the quality of the image given to the form must be high so that it contains more information and details, allowing the model to benefit from that information and details to improve the accuracy of analysis and classification, and it must With a large number of high-quality images in the training process, the model can learn a wide range of patterns and get a better ability to deal with new challenges, and it must also be taken into account that the size of the image is suitable for the model's ability to process efficiently, so we decided to visit the Egyptian museums and take images for statues we need standards. We considered that the images should be of high quality and clear enough, and they should be a variety of images representing different situations and styles that we want to teach the model to recognize. We varied the angles, positions, and lighting as much as possible to reduce contrast and noise caused by different lighting to obtain a variety of images.

Sample of Images as shown in figure 4.4

Figure 4.4:sample of images

4.1.4 Dataset preprocessing

Functions are used in image preprocessing.

- create_label (image_path)
  - ➢ It take path of image and make splitting to get name of image.
  - ➢ Transfer image name into binary array.
  - ➢ Return binary array for each image.
- ImageDataGenerator ()
  make Augmentation for each image
  - ➢ Rotation
  - ➢ Width Shifting
  - ➢ Height Shifting
  - ➢ Rescale
  - ➢ Shear
  - ➢ Zoom
  - ➢ Horizontal flip
  - ➢ Fill mode
  - ➢ Brightness
- create_train_data ()
  for each image:
  - ➢ Resize image to 200*200.
  - ➢ Covert color of image from BGR to RGB by using **cvtColor ().**
  - ➢ Create label for image.
  - ➢ Call ImageDataGenerator () to make Augmentation.
- create_test_data ()
  for each image:
  - ➢ Resize image to 200*200.
  - ➢ Covert color of image from BGR to RGB by using **cvtColor ().**
  - ➢ Create label for image.

4.2 Description of Software Tools Used

- **Dataset tools**

  - **Firebase**

    Firebase is a powerful tool for developers, as it provides a wide range of pre-built features and services that can be easily integrated into our application.

    Firebase helps us to save time and effort by handling many of the complex tasks associated with application development, such as data storage, authentication, and analytics. Additionally, firebase offers a scalable and reliable infrastructure that can support our application of any size.

- **Deep learning models tools**

  - **Google colab**

    Google colab is a cloud-based platform for developing and running python code in a jupyter notebook environment. It is a free service provided by google that allows users to write and run python code in their web browser, without requiring any local installation or setup, and we used it to create our deep learning model.

  - **Python**

    Python is a versatile, high-level programming language that is widely used for a variety of applications, including web development, data analysis, machine learning, and scientific computing and we used python to create our deep learning model, our deep fake model.

- **Desktop application tools**

  - **Visual studio code (vs code)**

    Visual studio code (vs code) is a popular code editor developed by Microsoft that used to create our desktop application using frameworks.

- **Tkinter**

  Tkinter is a Python library that provides a set of tools for creating graphical user interfaces (GUIs). It is a built-in library in Python and requires no additional installation, and we used it in our project to design desktop application.

• Mobile Application Tools

- **Android Studio**

  Android Studio is an integrated development environment (IDE) developed by Google that is designed specifically for developing Android applications. Android Studio provides a range of tools and features that can be used to create our mobile application using flutter.

- **Flutter**

  Flutter is an open-source mobile application development framework developed by Google that allows developers to build high-quality, cross-platform mobile applications for iOS and Android using a single codebase. Flutter provides a rich set of pre-built widgets and tools that can be used to create our android application.

• Other Tools

- **GitHub**

  GitHub is a web-based platform that provides a range of tools and features for software development, version control, and collaboration, we used it to manage and track changes to our code.

- **Replit API**

  Replit API is a platform that provides a set of tools and features for developers to create and manage API server.

4.3 Step Configuration (hardware)

The step configuration is to download the APK file of the application onto the mobile device. Once the APK file is downloaded, the next step is to install the application on the mobile device.

Installation can be initiated by tapping on the downloaded APK file, which will usually prompt the user to allow installation from unknown sources. Once this is allowed, the application installation process will begin as shown in figure 4.5, after a few seconds, the application will be successfully installed on the device.

The application is only available on the Android platform, it means that it is designed and developed specifically for mobile devices running on the Android operating system.

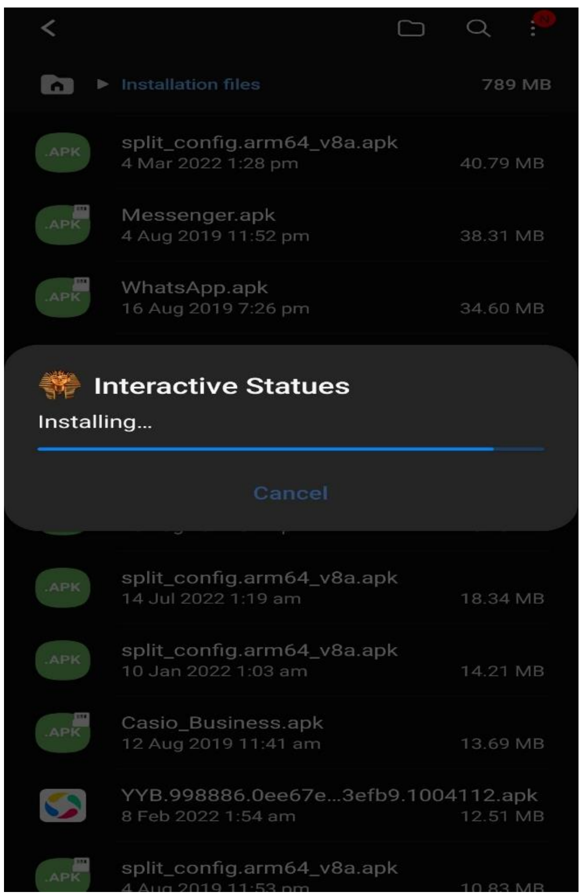

Figure 4.5: Installation of Application

4.4 Experimental and Results

4.4.1 Machine learning experiments

We use Support Vector Machine (SVM) to identify and classify the image of statue and we make many trials with using Augmentation and without Augmentation on our dataset as presented in table 4.1.

Table 4.1:SVM Results

	image size	Augmentation	Test accuracy
1	150	Yes - 4	0.5178
2	200	Yes - 4	0.3851
3	200	No	0.3398
4	228	No	0.4164
5	228	Yes - 2	0.5908

4.4.2 CNN experiments

- **LeNet-5:**

The LeNet-5 architecture consists of seven layers, including three convolutional layers, two pooling layers, and two fully connected layers [51].

We use LeNet-5 architecture to identify and classify the image of statue and we make many trials on our dataset as presented in table 4.2.

Table 4.2:LeNet-5 Results

	Epochs	Optimizer	Loss	Train Acc	Test Acc
1	5	Adam	3.8218	94.36%	34%
2	8	Adam	0.0675	98%	37%
3	15	Adam	0.0278	99.70%	35%

- **AlexNet:**

The AlexNet architecture consists of eight layers, including five convolutional layers and three fully connected layers [52].

We use AlexNet architecture to identify and classify the image of statue and we make many trials with using Augmentation and without Augmentation on our dataset as presented in table 4.3.

Table 4.3:AlexNet Results

	Epochs	Optimizer	Loss	Use Augmentation	Train Accuracy	Test Accuracy
1	5	Adam	21.5473	No	18%	15%
2	10	Adam	9.5495	No	17%	22%
3	15	Adam	6.0154	No	26%	33%
4	20	Adam	4.5187	No	30%	37%
5	30	Adam	3.0447	No	35%	46%
6	50	Adam	0.5469	Yes	84%	62%
7	100	Adam	0.0408	Yes	98%	64%
8	200	Adam	0.0411	Yes	98%	66%

- **Custom CNN**

Custom architecture consists of seven layers, including five convolutional layers and two fully connected layers.

We use Custom architecture to identify and classify the image of statue and we make many trials with using Augmentation and without Augmentation on our dataset as presented in table 4.4.

Table 4.4:Custom CNN Results

	Epochs	Optimizer	Loss	Train Accuracy	Test Accuracy	Use Augmentation
1	5	Adam	2.26098	18%	20%	No
2	10	Adam	1.17650	55.87%	47%	No
3	15	Adam	0.44931	72.74%	49%	No
4	20	Adam	0.8774	88.74%	52%	No
5	100	Adam	0.02644	99.3%	69.5%	Yes

- **VGG-16**

The VGG-16 architecture consists of 16 layers, including 13 convolutional layers and 3 fully connected layers [53].

We use VGG-16 architecture to identify and classify the image of statue and we make many trials with using Augmentation on our dataset as presented in table 4.5.

Table 4.5:VGG-16 Results

	Epochs	Optimizer	Loss	Train Accuracy	Test Accuracy
1	50	Adam	0.0430	98.78%	74%

- **VGG-19**

The VGG-19 architecture consists of 19 layers, including 16 convolutional layers and 3 fully connected layers [40].

We use VGG-19 architecture to identify and classify the image of statue and we make many trials with using Augmentation on our dataset as presented in table 4.6.

Table 4.6:VGG-19 Results

	image size	Augmentation	Learning rate	epochs	Train Accuracy	Validation accuracy	Test accuracy
1	200	no	0.001	30	0. 8045	0. 7213	0.6878
2	200	Yes - 4	0.0001	18	100	89.84	89.14
3	200	No	0.001	12	0.9660	0.7669	0.909
4	200	Yes - 4	0.001	10	0.9992	0.9638	0.9638
5	200	Yes - 4	0.001	30	0.9884	0.9140	0.914
6	200	Yes - 6	0.001	9	0.9266	0.8652	0.8778
7	200	Yes - 4	0.0001	30	0.1000	0.9368	0.9366516
8	200	No	0.0001	10	0.1000	0.9515	0.8757
9	200	No	0.001	7	0.9898	0.9689	0.9276
10	200	Yes - 2	0.001	30	0.9981	0.9619	0.96186

4.4.3 Results of Deep learning models

We achieved a remarkable accuracy rate of 97% by using VGG-19. This high accuracy indicates the model's proficiency in identifying different statutes and extracting relevant features as shown in table 4.7.

Table 4.7:Summary of Results

Model	Augmentation	Accuracy
LeNet-5	No	37%
Custom	Yes	69.5%
Custom	No	52%
AlexNet	Yes	66%
AlexNet	No	46%
VGG-16	Yes	74%
SVM	No	41.64%
SVM	Yes	59.08%
VGG-19	No	92.76%
VGG-19	Yes	96.78%

4.4.3 User-Friendly Mobile Interface

The mobile application's user interface, built using the Flutter framework, was designed with simplicity and ease of use in mind. The application offered smooth navigation, quick loading times, and responsiveness across different screen sizes and devices. The visually appealing interface, accompanied by intuitive symbols and a well-coordinated color palette, contributed to an enjoyable user experience.

4.4.4 Desktop Application and Database Management

The desktop application provided administrators with comprehensive control over the database. They could easily create, edit, delete, and add videos of Talking Statues. The application, developed using the Tkinter library, offered a user-friendly graphical interface with various elements and functions, enabling efficient management of the content.

4.4.5 Deep Fake Experiments and Results

The first-order motion model played a crucial role in the application by enabling the replacement of faces in the driving video with the likeness of the desired statue. This advanced technology allowed for the creation of videos where the statue's face appeared to come to life, delivering historical information and engaging in conversations. By seamlessly integrating the statue's face into the video, the application was able to produce realistic and high-quality results, showcasing the power of deep fake technology.

These generated videos demonstrated the immense potential of deep fake technology in enhancing interactive experience. By utilizing the first-order motion model, the application achieved a level of realism that captivated users and provided an immersive and engaging experience. The ability to animate statues and make them appear to speak opened up new possibilities for storytelling and historical education, showcasing the exciting capabilities of this technology in creating compelling visual content.

Measuring the accuracy of the first-order motion model typically involves evaluating the quality of the generated videos and assessing how well they align with the desired outcomes, there are factors can influence accuracy of model :

- Video Processing: The quality and characteristics of the input video can impact the accuracy of the model. Factors such as video resolution, frame rate, lighting conditions, and camera stability can affect the model's ability to track facial movements accurately. Higher-quality videos with clear and consistent facial details are more likely to yield better results in terms of accuracy. Additionally, any preprocessing steps applied to the video, such as noise reduction or stabilization, can help improve the accuracy of the model by providing cleaner input data.

- Image Processing: The accuracy of the first-order motion model can also be influenced by image processing techniques applied during the facial replacement process. These techniques may involve image enhancement, alignment, or blending to ensure a seamless integration of the new face onto the original video frames. The effectiveness of these image processing steps can impact the accuracy of the generated videos. Careful attention should be given to preserving facial details, maintaining naturalness, and avoiding artifacts or distortions that could affect the accuracy of the final output.

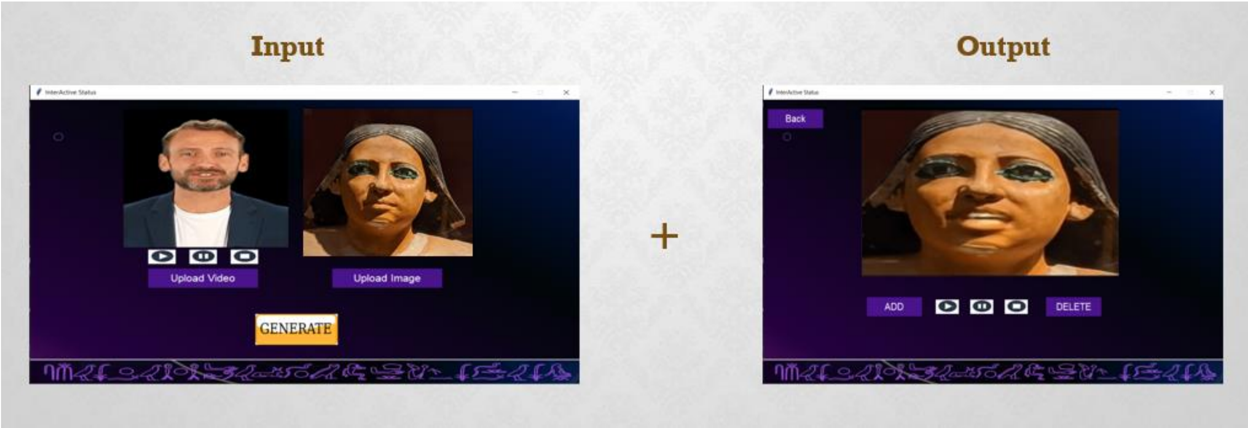

Figure 4.6:first order motion model result

5) Chapter 5: Run the Application

5.1 Run Desktop Application

5.1.1 Main Screen in Desktop Application:

The main screen for a desktop application presents administrators with two options to choose from as shown in figure 5.1.

The first option, "Create Deepfake Videos," is intended to assist the administrator in creating a new deep-fake video for a statue. When the administrator selects this option, they will be taken to another screen.

The second option on the main screen is to "Manage Database." This option is designed to provide users with access to all the statues data previously saved in the application's database. Users who choose this option will be able to manage the database by adding or deleting data as needed.

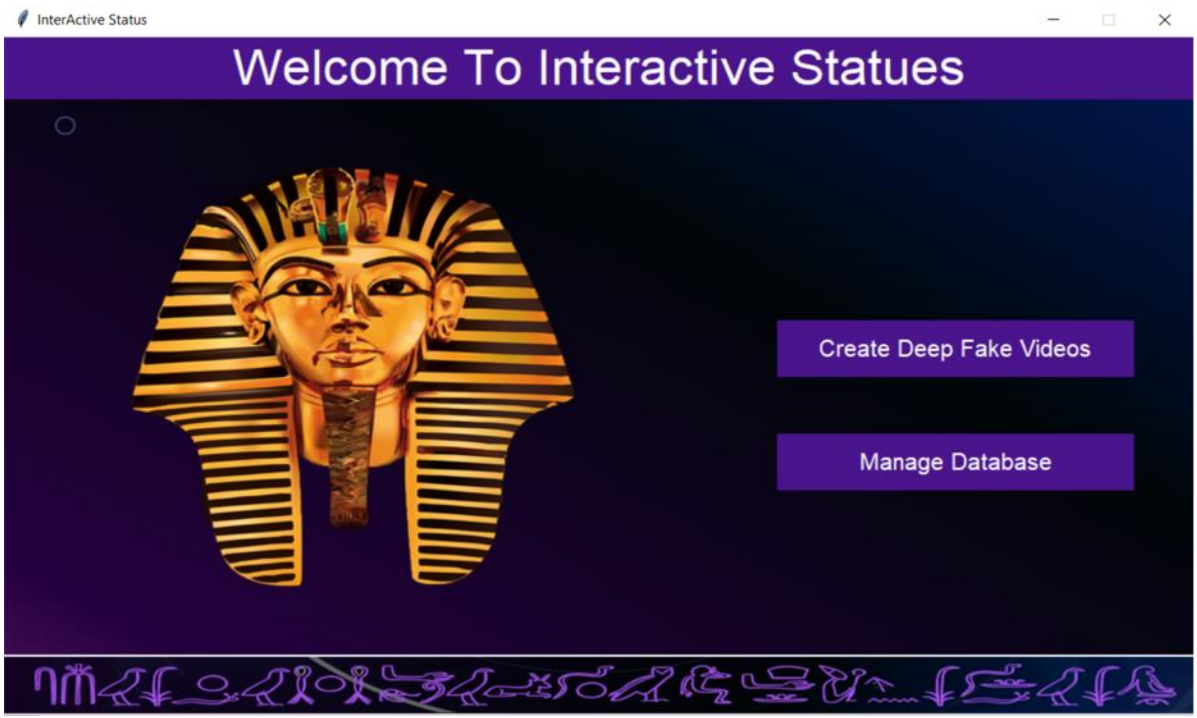

Figure 5.1:Main Screen in Desktop Application

5.1.2 Create Deepfake video Screen:

The "Create Deepfake Video" screen is displayed when the admin presses the "Create Deepfake Videos" button, which is located on the main screen. The main screen is the first screen that appears when the admin logs in to the system and displays several options for creating and managing deepfake videos.

When the "Create Deepfake Videos" button is clicked, the system navigates to the "Create Deepfake Video" screen. The admin can begin selecting the driving video and source image for their deep-fake creation as shown in figure 5.2. The admin has the option to select a driving video for a person speaking as the statue and describing themselves and has the option to select a source image of the statue they want to create a deep-fake for. The source image will be animated according to the motion of the driving video, creating a realistic-looking deep-fake video.

Furthermore, the admin also has the option to control the driving video by playing or pausing it. he can ensure that the video runs smoothly and matches their desired pace. Once the admin has selected their desired video and image, he can click on the "GENERATE" button to start the creation of the fake video as shown in figure 5.2.

Once the "GENERATE" button has been clicked, the output screen will display the final deep fake video of the statue. The admin can play or pause the video to review it. If admin is satisfied with the output, he can then click on the "Add" button, which is linked to a firebase repository, to store the video as shown in figure 5.3.

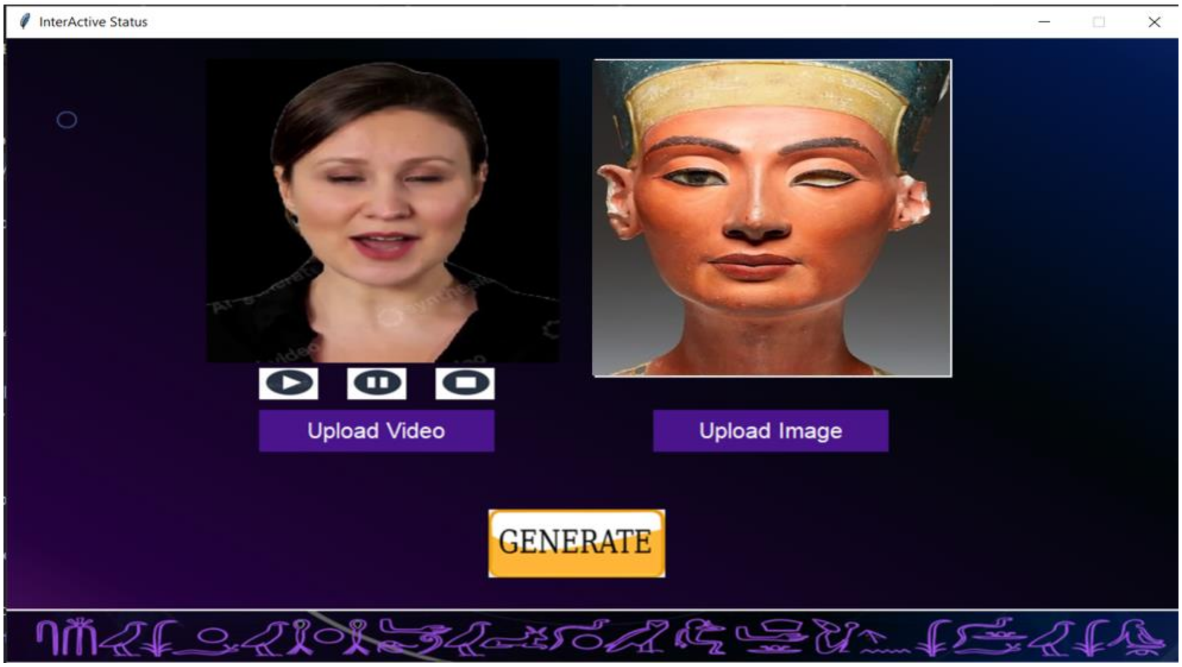

Figure 5.2:Create Deepfake video Screen in Desktop Application.

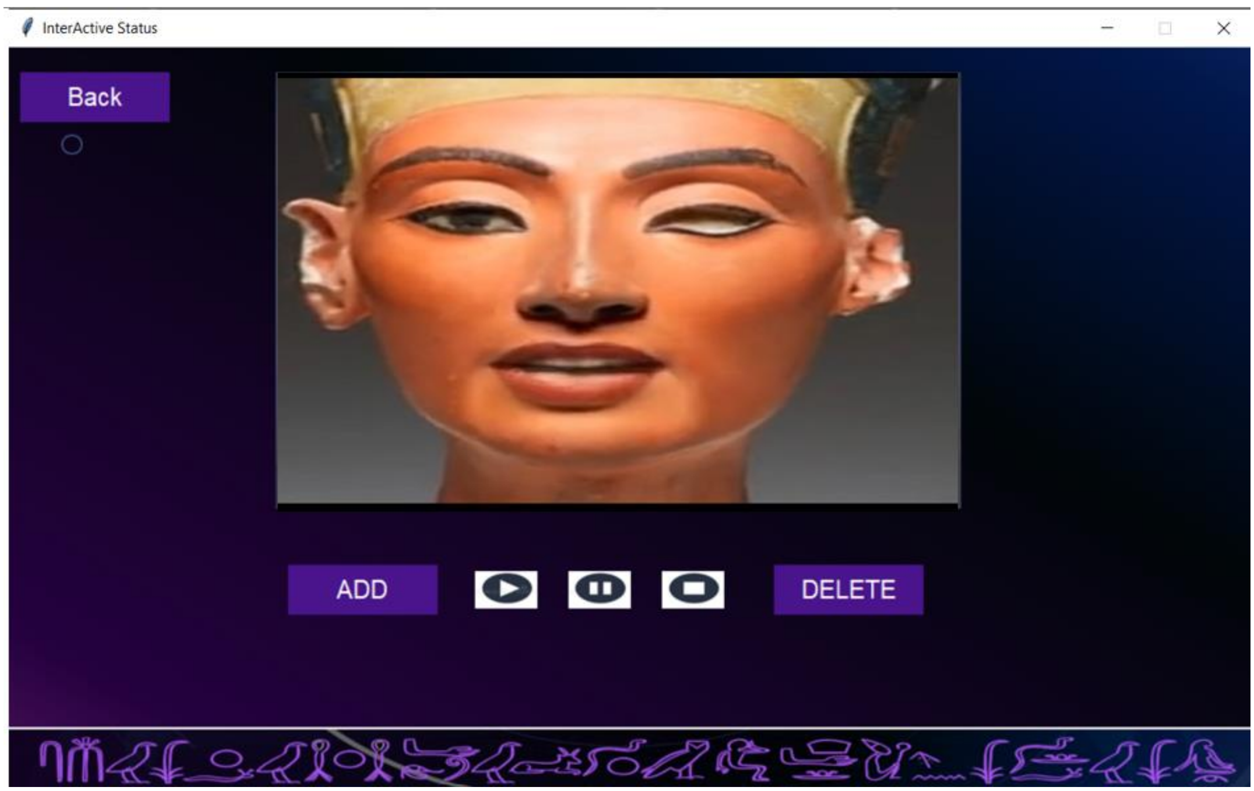

Figure 5.3:Output Screen in Desktop Application

5.1.3 Manage Database Screen:

The "Manage Database Screen" is displayed when the admin presses the "Manage Database" button, which is located on the main screen. The main screen is the first screen that appears when the admin logs in to the system and displays several options for creating and managing deepfake videos.

The "Manage Database Screen" would be a screen where the admin can perform various operations on the database. This screen would typically include options to search and delete video from the database.

To search for data, the admin can enter the statue name and choice the statue name from radio buttons as shown in figure 5.4 to perform a search, which will display corresponding data. The displayed data would include the statue deep-fake video.

After displaying the deep-fake video of statue, the admin can review the video as required by watch the video, if the user wants to delete the status, he can simply click on the "Delete" button to remove it from the database as shown in figure 5.5.

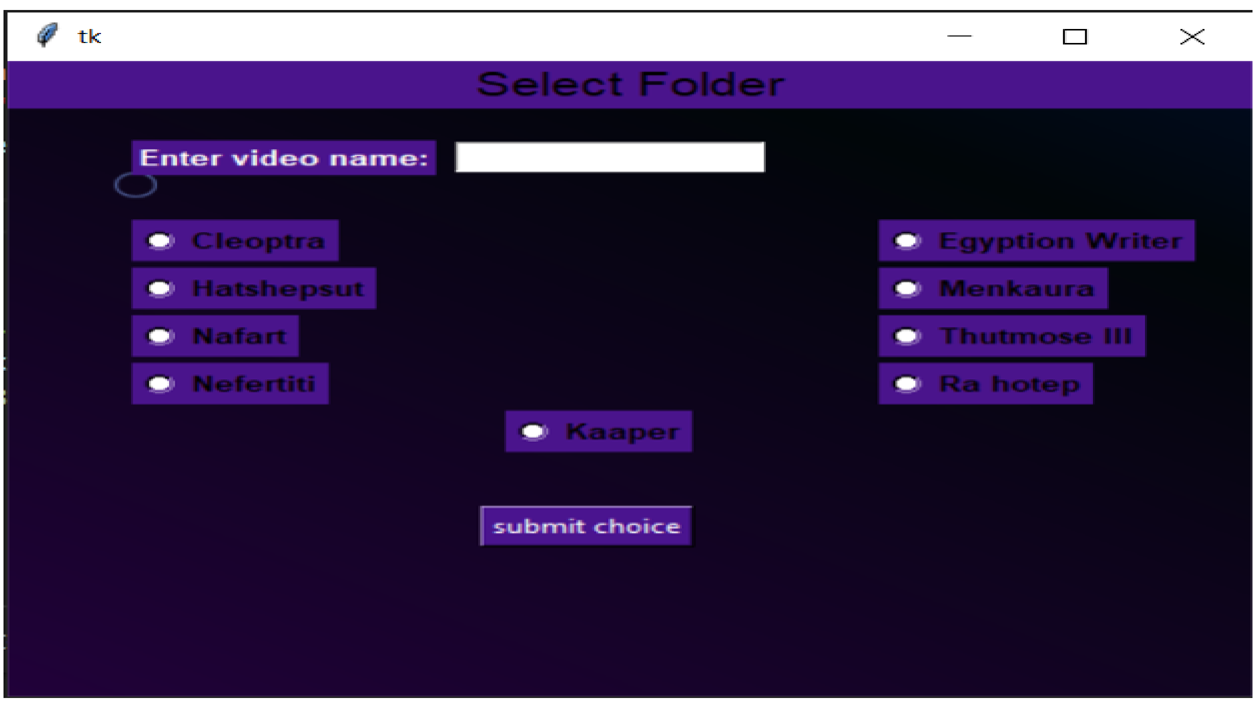

Figure 5.4:Search Screen in Desktop Application

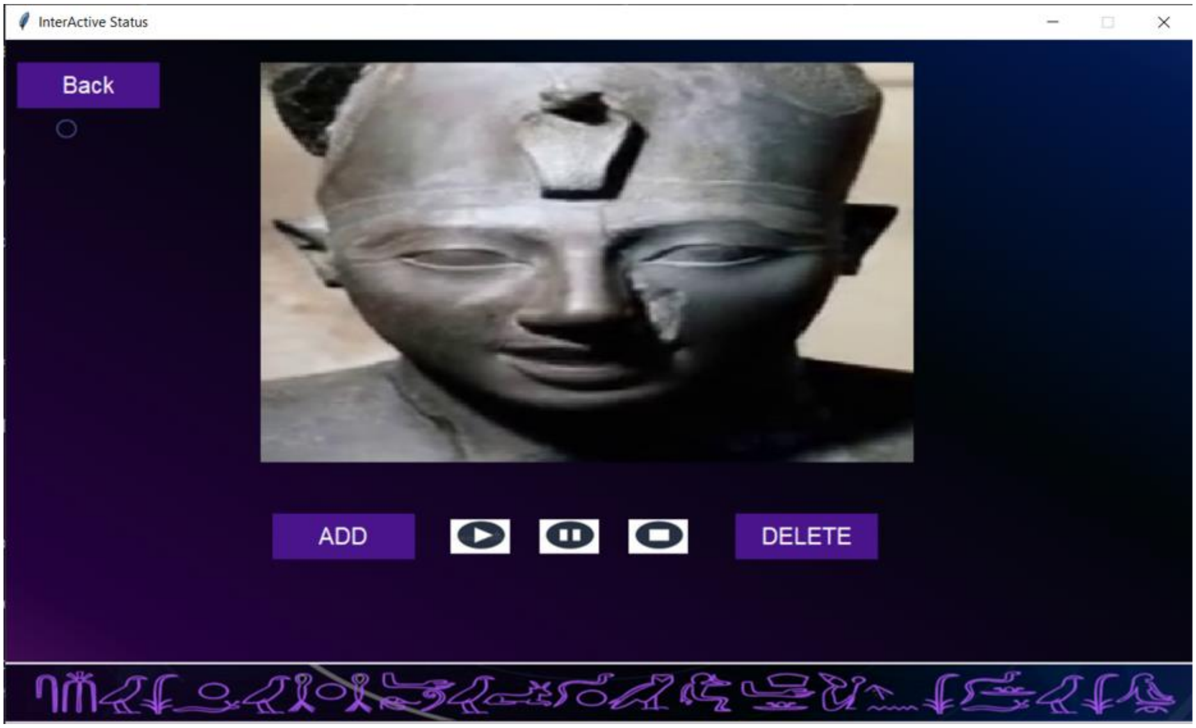

Figure 5.5:Watch Video Screen in Desktop Application

5.2 Run Mobile Application

Interactive Statues is a mobile application that has been recently developed for Android devices. It aims to provide users with an immersive and interactive experience of knowing historical information about statues.

5.2.1 Splash Screen in Mobile Application:

A Splash Screen is an integral part of our application and is the first screen that appears when the user launches the application. The splash screen is a graphical representation of the application 's branding and provides a brief introduction and visual appeal to the users and is an animation screen as shown in figure 5.6.

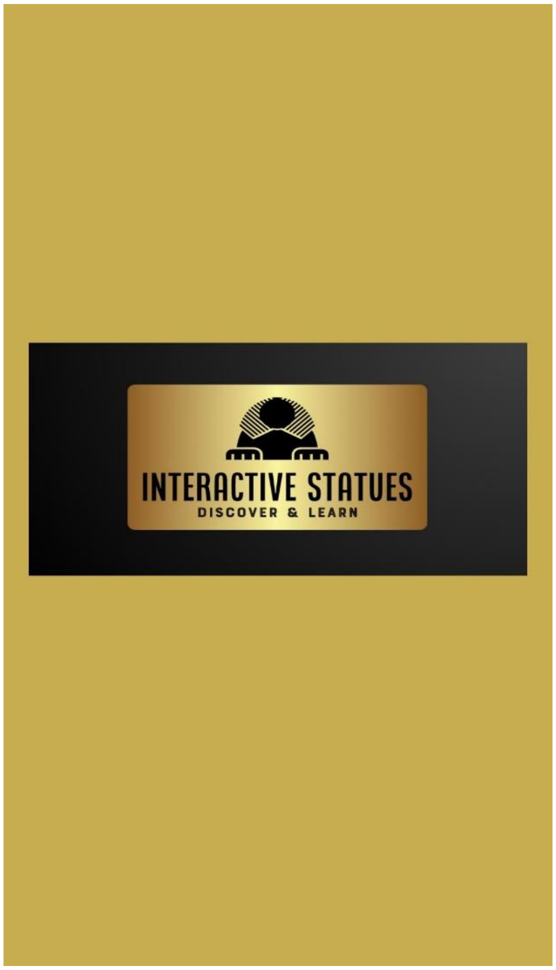

Figure 5.6:Splash Screen in Mobile Application

5.2.2 Onboarding screens in Mobile Application:

Onboarding screens are an essential component of our application that aims to provide new users with a seamless and immersive experience.

Onboarding screens that are only displayed during the first opening of an Android application are crucial for creating a seamless and user-friendly experience.

Onboarding screens of an Android application offer a great opportunity to introduce the users to the application's features and encourage them to make full use of its potential. For instance, these screens can show the user how to pick or discover an image of a statue while taking spatial restrictions into account. Additionally, the onboarding screens can highlight the benefits of the app and give a general overview of what the app is all about.

Moreover, the onboarding screens can demonstrate how to make use of the app's various functionalities, such as interactive chat. The onboarding screens ensure that the user understands the app's functionalities, how to use them, and the benefits they provide.

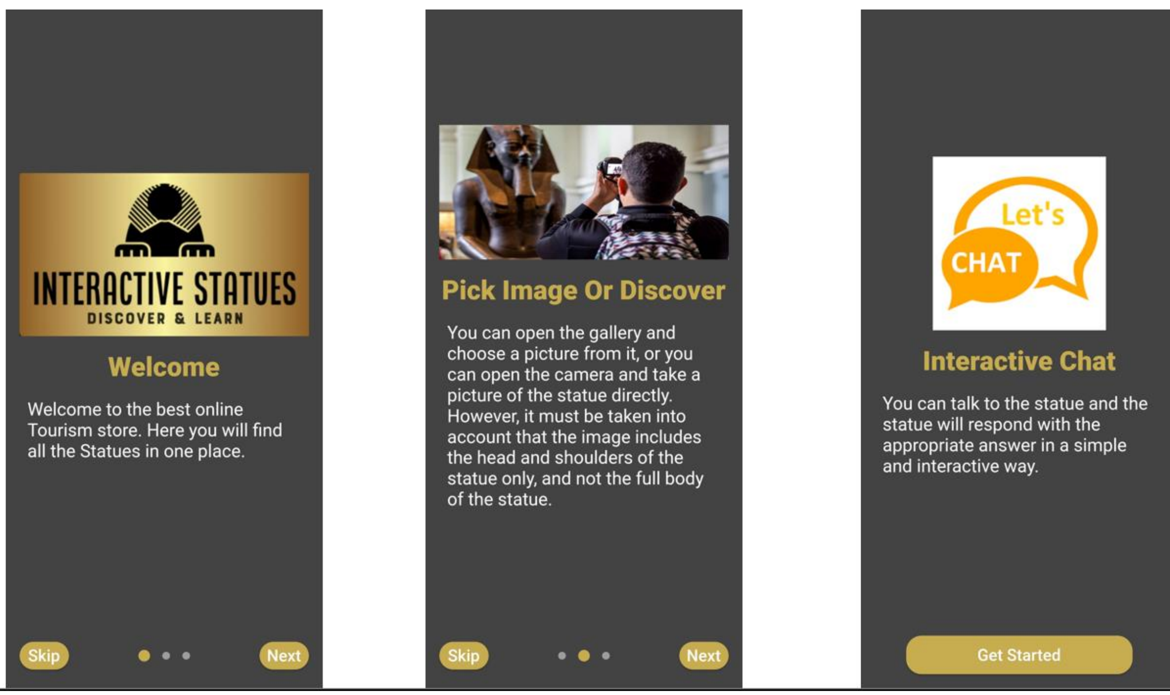

Figure 5.7:Onboarding Screens in Mobile Application

5.2.3 Home Screen in Mobile Application:

A Home Screen that includes all statues offers the user a convenient way to browse through the application's collection of statues. The Home Screen displays images of all statues in a grid or list format, allowing user to easily scroll through and locate the statue they are interested in as shown in figure 5.8. By clicking on any statue on the Home Screen, user can be directed to the statue's specific screen, where they can view more details about the statue, and provide "lets chat" button that allows user to go to chat screen.

The Home Screen can also include icon One and icon Two on the Home Screen of an application that enables users to identify statues they come across in real life as shown in figure 5.8.

Specifically, by clicking on icon One, users can use their device's camera to capture an image of the statue. The application would then navigate to an Edit Screen, where users can crop and edit the image they just captured as shown in figure , and when users click on icon(3) in figure 5.9,The application can use image recognition technology to identify the statue and navigate users to a specific Statue Screen that offers more information about the identified statue and provide "lets chat" button.

Icon two functions similarly to icon One, allowing users to select an image of the statue they previously captured from their device's gallery and the application would then navigate to an Edit Screen, where users can crop and edit the image they just captured as shown in figure and when users click on icon(3) in figure 5.9 ,The Application use image recognition technology to identify the statue and navigate to its specific Statue Screen.

The Home Screen icons are a useful feature that allows users to identify any statue easily and swiftly they come across, enriching their understanding and appreciation of the statue.

Figure 5.8:Home Screen in Mobile Application

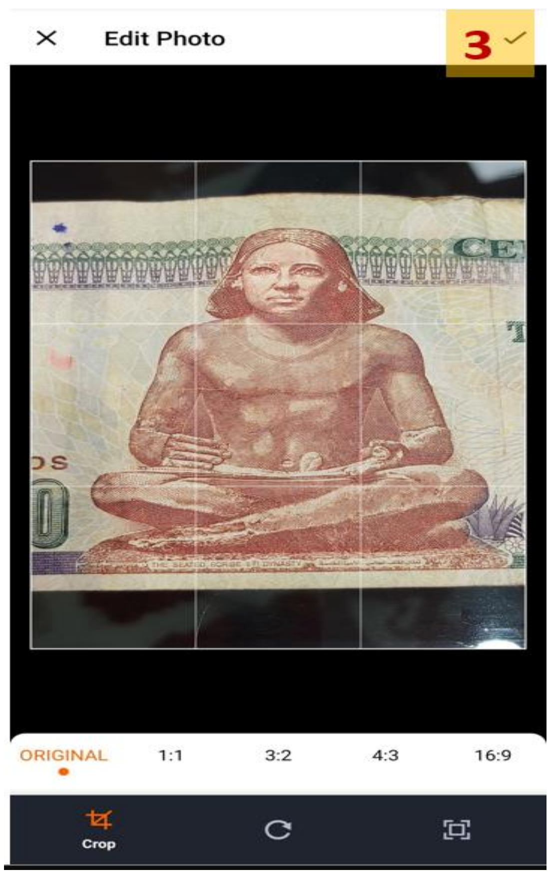

Figure 5.9:Edit Screen in Mobile Application

5.2.4 Statue Screen in Mobile Application:

The Statue Screen in our application offers detailed information about the identified statue. The screen includes images of the statue from different angles, allowing users to appreciate the statue's intricate design details. The screen also features a description of the statue, including its history as shown in figure 5.10, This wealth of information enhances users' understanding and appreciation of the statue and its significance.

Moreover, the Statue Screen includes a "Let's Chat" button that enables users to discuss with the statue, when clicked, this button navigates users to the Chat Screen, where users can engage in a real-time conversation with statue.

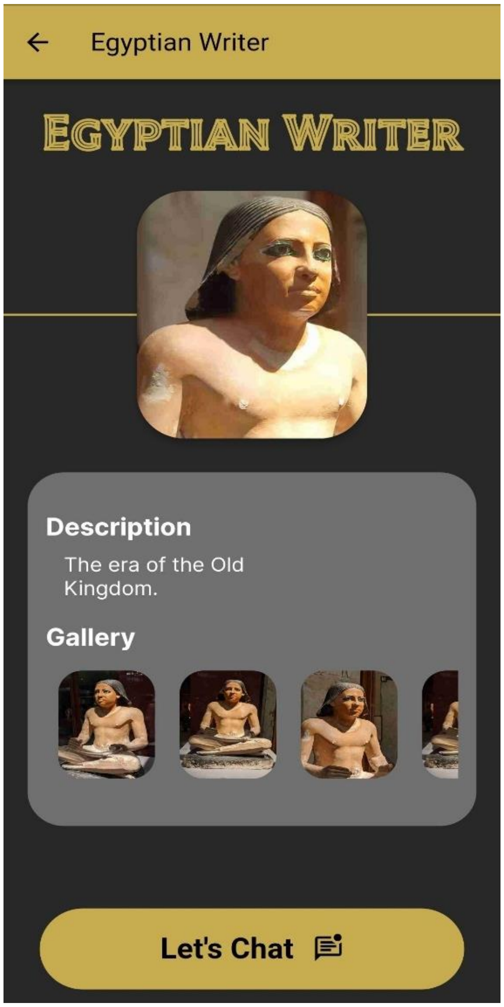

Figure 5.10:Statue Screen in Mobile Application

5.2.5 Chat Screen in Mobile Application:

The Chat Screen in the mobile application is a user-friendly feature that enables users to ask questions about the statues they are interested in. The Chat Screen provides two options for users to ask questions.

The first option is to click on the pre-set questions that are shown in the figure 5.11. This feature is ideal for users who want to know more about specific aspects of the statue. These pre-set questions are curated carefully to provide users with the most relevant and informative answers about the statue.

The second option for users is to utilize the microphone button as shown in figure 5.11, which allows them to record their own questions, speaking to them aloud in their own voice. The application then receives a fake video response that provides answers to the user's question. This feature offers a more personalized and interactive experience for users, giving them the chance to have their questions answered in real-time through a conversational format. Overall, the Chat Screen is a valuable feature in the app, offering users two unique options for asking their questions, making the exchange of knowledge and information about the statues fun and enjoyable.

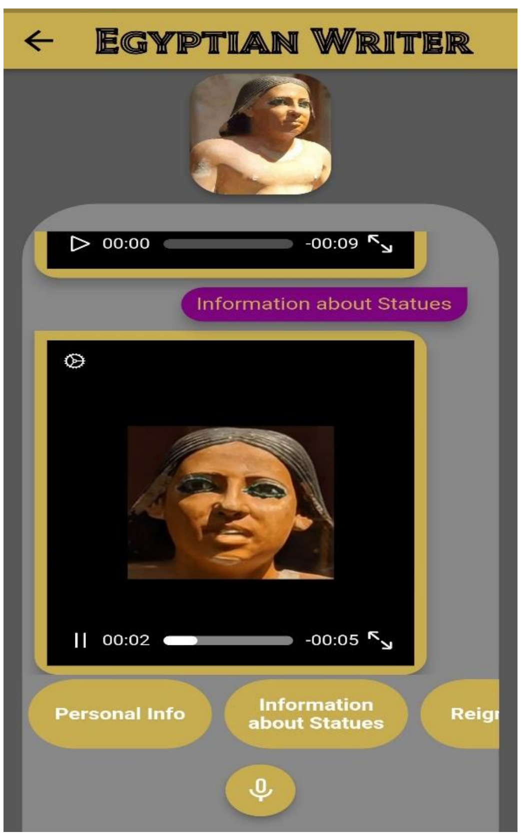

Figure 5.11:Chat Screen in Mobile Application

6) Chapter 6: Conclusion and Future Work

6.1 Conclusion

In conclusion, our research endeavors have led to the development of a mobile application that seeks to revolutionize the tourist experience in Egypt and beyond. By harnessing the power of deep learning, machine language, and the Flutter Framework, we have created a comprehensive and interactive platform that provides tourists, researchers, and students with easy access to information about ancient Egyptian kings and queens. This application not only addresses the challenges faced by tourists, such as the scarcity of tourist guides and inflated trip prices but also aims to enhance cultural exploration, promote tourism, and increase awareness of the rich Egyptian civilization.

Through the implementation of deep learning models, including the VGG-19 convolutional neural network for image recognition and classification, as well as the first-order motion model for generating realistic talking statues, our application offers users a unique and immersive experience. By engaging in conversations with the statues, users can gain in-depth knowledge, ask questions, and receive video responses, thus bringing the statues to life and enabling a deeper understanding of their historical significance.

The proposed methodology encompasses a mobile application for users, a desktop application for administrators and database managers, a dataset comprising historical information and textual data from reputable sources, and robust deep-learning models for image recognition and video generation. Each component has been carefully designed and implemented to ensure a seamless and user-friendly experience, facilitating efficient interaction with the application.

As we conclude this study, we remain committed to advancing the boundaries of technology-enabled tourism and look forward to the future, where our application can continue to inspire and educate individuals from all corners of the globe.

6.2 Future Work

While our mobile application has made significant strides in revolutionizing the tourist experience and promoting cultural exploration, there are several areas that can be further explored and improved upon in future work.

These areas include:

- **Comprehensive Coverage of the Egyptian Civilization**:
  Our initial focus has been on providing information and interactive experiences related to the statues of kings and queens in Egypt. However, there is still a vast number of statues and historical artifacts spread across the country. Future work should involve expanding the coverage to include a broader range of statues, monuments, and archaeological sites, enabling users to explore the entire Egyptian civilization. This would require extensive fieldwork, collaboration with museums and archaeological institutions, and the collection of high-quality photos and data to create a comprehensive database.

- **Expansion to Other Ancient Civilizations:** As we mentioned in our conclusion, one of our future goals is to expand the application beyond Egypt and incorporate statues and artifacts from various ancient civilizations. This would involve conducting research, gathering historical data, and developing deep learning models specific to each civilization. By broadening the scope of the application, we can create a truly global platform for cultural exploration.

- **Continuous Improvement of Deep Learning Models:** While our current deep learning models, such as the VGG-19 convolutional neural network and first order motion model, have yielded promising results, there is always room for improvement. Future work can focus on fine-tuning the models, increasing their accuracy, and exploring new architectures and techniques that could further enhance image recognition, video generation, and natural language processing capabilities.

- **Multi-Language Support:** To cater to a wider audience, it would be valuable to expand the application's language support beyond its current offerings. This would involve translating the textual dataset, implementing multi-language natural language processing capabilities, and ensuring the smooth functioning of the application across different languages. By accommodating diverse linguistic backgrounds, the application can attract a larger user base and facilitate cross-cultural communication.

- **User Feedback and Iterative Improvements:** Gathering feedback from users is crucial for understanding their needs, preferences, and potential issues they may encounter while using the application. Conducting user surveys, analyzing user behavior data, and actively seeking user feedback can provide valuable insights for iterative improvements. Incorporating a feedback loop into the development process will allow for continuous refinement and enhancement of the application based on user experiences and requirements.

- **Collaboration with Tourism Authorities and Institutions:** Collaborating with tourism authorities, museums, and academic institutions can contribute to the authenticity, accuracy, and relevance of the application's content. By establishing partnerships and obtaining access to curated databases and resources, we can ensure that the application remains up to date with the latest research, historical findings, and cultural preservation efforts. Such collaborations can also provide opportunities for joint marketing initiatives and increased visibility for the application.

In conclusion, future work should focus on expanding the application's coverage to include other ancient civilizations, integrating augmented reality features, improving deep learning models, enhancing multi-language support, gathering user feedback, and fostering collaborations with relevant stakeholders. By pursuing these avenues, we can continue to innovate and evolve the application, making it an indispensable tool for cultural exploration, tourism, and historical education in the years to come.

References

[1] M. Naguib, "Ahramonline," 31 december 2022. [Online]. Available: https://english.ahram.org.eg/News/482899.aspx. [Accessed 25 march 2023].

[2] M. Sládek, M. K. Röschová, V. Adámková and D. H. &. A. Sumová, "Martin Sládek, Michaela Kudrnáčová Röschová, Věra Adámková, Dana Hamplová & Alena Sumová," vol. 10, p. 18, 2020.

[3] J. Wu, K. Feng, X. Chang and T. Yang, "A Forensic Method for DeepFake Image based on Face Recognition," in *HPCCT & BDAI 2020: 2020 4th High Performance Computing and Cluster Technologies Conference & 2020 3rd International Conference on Big Data and Artificial Intelligence*, Qingdao , 2020.

[4] Y. Zhou, X. Han, E. Shechtman, J. Echevarria, E. Kalogerakis and D. Li, "MakeltTalk: speaker-aware talking-head animation," *MakeItTalk: Speaker-Aware Talking-Head Animation,* vol. 39, no. 6, p. 15, 2020.

[5] T. T. Nguyen, Q. V. H. Nguyen, D. T. Nguyen, D. T. Nguyen, T. Huynh-The, S. Nahavandi, T. T. Nguyen, Q.-V. Pham and C. M. Nguyen, "Deep Learning for Deepfakes Creation and Detection," *Deep Learning for Deepfakes Creation and Detection: A Survey,* p. 19, 2022.

[6] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci and N. Sebe, "First Order Motion Model for Image Animation," in *Neural Information Processing Systems (NeurIPS 2019)*, Vancouver, 2019.

[7] M. Lal, K. Kumar, R. H. Arain and A. Maitlo, "Face Recognition," *Study of Face Recognition Techniques: A Survey,* vol. 9, no. 6, p. 8, 2018.

[8] R. Chauhan, K. Kumar and G. R. Joshi, "Convolutional Neural Network (CNN) for Image Detection and Recognition," in *Secure Cyber Computing and Communication (ICSCCC)*, 2018.

[9] H.-R. Chou, J.-H. Lee, Y.-M. Chan and C.-S. Chen, "Data-specific Adaptive Threshold for Face Recognition and Authentication," in *IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2018.

[10] M. Hussain, J. J. Bird and D. R. Faria, "A Study on CNN Transfer Learning for Image Classification," in *Advances in Computational Intelligence Systems: Contributions Presented at the 18th UK Workshop on Computational Intelligence*, 2018.

[11] J. FRANKENFIELD, "Investopedia-Artificial Intelligence," 6 July 2022. [Online]. Available: https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp. [Accessed 22 march 2023].

[12] V. Kanade, "spiceworks-Artificial Intelligence," 14 March 2022. [Online]. Available: https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-ai/. [Accessed 21 march 2023].

[13] careers.noreply, "expert.ai," 14 March 2022. [Online]. Available: https://www.expert.ai/blog/machine-learning-definition/. [Accessed 25 march 2023].

[14] V. Kanade, "spiceworks," 30 August 2022. [Online]. Available: https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-ml/. [Accessed 23 March 2023].

[15] B. Mahesh, "Machine Learning Algorithms," *International Journal of Science and Research,* vol. 9, no. 1, p. 6, 2019.

[16] S. Ray, "A Quick Review of Machine Learning Algorithms," in *Machine Learning, Big Data, Cloud and Parallel Computing (Com-IT-Con)*, India, 2019.

[17] M. McGregor, "freecodecamp-SVM," 1 july 2020. [Online]. Available: https://www.freecodecamp.org/news/svm-machine-learning-tutorial-what-is-the-support-vector-machine-algorithm-explained-with-code-examples/. [Accessed 3 April 2023].

[18] V. Kanade, "spiceworks-svm," 29 September 2022. [Online]. Available: https://www.spiceworks.com/tech/big-data/articles/what-is-support-vector-machine/. [Accessed 27 March 2023].

[19] "ibm-deeplearning," [Online]. Available: https://www.ibm.com/topics/deep-learning. [Accessed 27 march 2023].

[20] K. Reyes, "simplilearn-what-is-deep-learning," 12 february 2023. [Online]. Available: https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-deep-learning. [Accessed 23 march 2023].

[21] T. W. Smith and S. A.Colby, "Teaching for Deep Learning," vol. 80, no. 5, p. 7, 2007.

[22] M. Mandal, "analyticsvidhya-Introduction to Convolutional Neural Networks (CNN)," 1 May 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/. [Accessed 1 April 2023].

[23] "deci.ai-Convolutional Neural Network (CNN)," [Online]. Available: https://deci.ai/deep-learning-glossary/convolutional-neural-network-cnn/. [Accessed 1 April 2023].

[24] K. Ahmad, "makeuseof-convolutional-neural-network-," 29 December 2022. [Online]. Available: https://www.makeuseof.com/convolutional-neural-network-explained/. [Accessed 27 march 2023].

[25] K. Gandharv, "indiaai-applications-of-convolution-neural-network," 29 June 2022. [Online]. Available: https://indiaai.gov.in/article/top-5-applications-of-convolution-neural-network. [Accessed 28 march 2023].

[26] "intellipaat-convolution-neural-network," [Online]. Available: https://intellipaat.com/blog/tutorial/artificial-intelligence-tutorial/convolution-neural-network/. [Accessed 29 march 2023].

[27] M. M.-C. N. Networks, "towardsdatascience," 26 Aug 2020. [Online]. Available: https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939. [Accessed 28 March 2023].

[28] "ibm-convolutional-neural-networks," [Online]. Available: https://www.ibm.com/topics/convolutional-neural-networks. [Accessed 28 March 2023].

[29] J. Sawtell-Rickson, "builtin-deepfake," 23 January 2023. [Online]. Available: https://builtin.com/machine-learning/deepfake. [Accessed 2 April 2023].

[30] D. Johnson, "businessinsider-deepfake," 10 August 2022. [Online]. Available: https://www.businessinsider.com/guides/tech/what-is-deepfake. [Accessed 2 April 2023].

[31] "gao-deepfake," U.S. Government Accountability Office, 20 October 2020. [Online]. Available: https://www.gao.gov/blog/deconstructing-deepfakes-how-do-they-work-and-what-are-risks. [Accessed 2 April 2023].

[32] N. Barney, "techtarget-deepfake," [Online]. Available: https://www.techtarget.com/whatis/definition/deepfake. [Accessed 2 April 2023].

[33] E. NANOU, "makeuseof-deepfake," 25 October 2022. [Online]. Available: https://www.makeuseof.com/how-do-deepfakes-work-and-who-is-using-them/. [Accessed 2 April 2023].

[34] "knowledgenile-deepfake," [Online]. Available: https://www.knowledgenile.com/blogs/applications-of-deepfake-technology-positives-and-dangers/. [Accessed 3 April 2023].

[35] "rubikscode-first order motion," 31 May 2021. [Online]. Available: https://rubikscode.net/2021/05/31/create-deepfakes-in-5-minutes-with-first-order-model-method/. [Accessed 3 April 2023].

[36] "Software Architecture & Design Introduction," [Online]. Available: https://www.tutorialspoint.com/software_architecture_design/introduction.htm. [Accessed 13 May 2023].

[37] E. Ayan, B. Karabulut and H. M. Ünver, "CNN model Architecture," *Diagnosis of Pediatric Pneumonia with Ensemble of Deep Convolutional Neural Networks in Chest X-Ray Images,* p. 17, 2021.

[38] P. Elangovan and M. K. Nath, "CNN model Architecture," *En-ConvNet: A novel approach for glaucoma detection from,* p. 15, 2021.

[39] A. Bagaskara and M. Suryanegara, "Evaluation of VGG-16 and VGG-19 Deep Learning Architecture for Classifying Dementia People," in *International Conference of Computer and Informatics Engineering (IC2IE)*, Depok, Indonesia, 2021.

[40] D. I. Sec., "VGG-19 Convolutional Neural Network," 6 March 2021. [Online]. Available: https://blog.techcraft.org/vgg-19-convolutional-neural-network/. [Accessed 25 June 2023].

[41] Y. Yu, K. Adu, N. Tashi, P. Anokye, X. Wang and M. A. Ayidzoe, " Activation Functions," *RMAF: Relu-Memristor-Like Activation Function for Deep Learning,* vol. 8, p. 15, 2020.

[42] J. Wu, X.-Y. Chen, H. Zhang, L.-D. Xiong and H. Lei, "Hyperparameter Optimization," *Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimization,* vol. 17, no. 1, pp. 26-40, 2019.

[43] L. Yang and A. Shami, *On hyperparameter optimization of machine learning algorithms: Theory and practice,* vol. 415, p. 69, 2019.

[44] A. Kaushik, "opengenus," [Online]. Available: https://iq.opengenus.org/vgg19-architecture/. [Accessed 24 June 2023].

[45] A. Rosebrock, "pyimagesearch," 20 March 2017. [Online]. Available: https://pyimagesearch.com/2017/03/20/imagenet-vggnet-resnet-inception-xception-keras/. [Accessed 24 June 2023].

[46] M. Huh, P. Agrawal and A. A. Efros, "What makes ImageNet good for transfer learning?," *What makes ImageNet good for transfer learning?,* p. 10, 2016.

[47] Y. Zhang, Y. Zhao, Y. Wen, Z. Tang, X. Xu and M. Liu, "Facial Prior Based First Order Motion Model for Micro-expression Generation," in *Proceedings of the 29th ACM International Conference on Multimedia.* , china, 2021.

[48] A. Dima, S. Lukens, M. Hodkiewicz, T. Sexton and M. P. Brundage, " natural language processing," *Adapting natural language processing for technical text,* vol. 3, no. 2, p. 11, 2021.

[49] C. Khanna, "Text pre-processing: Stop words removal using different libraries," 10 Feb 2021. [Online]. Available: https://towardsdatascience.com/text-pre-processing-stop-words-removal-using-different-libraries-f20bac19929a. [Accessed 25 june 2023].

[50] K. Pykes, "Stemming and Lemmatization in Python," Feb 2023. [Online]. Available: https://www.datacamp.com/tutorial/stemming-lemmatization-python. [Accessed 25 June 2023].

[51] M. Kayed, A. Anter and H. Mohamed, "Classification of Garments from Fashion MNIST Dataset Using CNN LeNet-5 Architecture," in *International Conference on Innovative Trends in Communication and Computer Engineering (ITCE)*, Aswan, 2020.

[52] S. Sudhakara, A. J. Prabhu, Ramachandran, V. Priya, R. Logeshd and Subramaniyaswamy, " AlexNet architecture," *Images super-resolution by optimal deep AlexNet architecture for medical application,* vol. 39, p. 6, 2020.

[53] R. Thakur, "Step by step VGG16 implementation in Keras for beginners," 6 Augest 2019. [Online]. Available: https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c. [Accessed 15 May 2023].

[54] A. Oppermann, "builtin-How Does deeplearning Work," 2 May 2022. [Online]. Available: https://builtin.com/machine-learning/what-is-deep-learning. [Accessed 20 march 2023].

[55] "javatpoint-Machine Learning," [Online]. Available: https://www.javatpoint.com/machine-learning. [Accessed 21 march 2023].

[56] K. Ahmad, "makeuseof-What Is a Convolutional Neural Network (CNN)," 29 DEC 2022. [Online]. Available: https://www.makeuseof.com/convolutional-neural-network-explained/. [Accessed 27 march 2023].

[57] "mathworks-Deep Learning," [Online]. Available: https://www.mathworks.com/discovery/deep-learning.html. [Accessed 24 March 2023].

[58] M. Lal, K. Kumar, R. Hussain, Arain, A. Maitlo, S. A. Ruk and H. S. , "Study of Face Recognition Techniques," *Advanced Computer Science and Applications,* vol. 9, no. 6, p. 8, 2018.

[59] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci and N. Sebe, "First Order Motion Model for Image Animation," in *Neural Information Processing Systems (NeurIPS 2019)*, Vancouver, 2019.

[60] M. Hussain, J. J. Bird and D. R. Faria, "A Study on CNN Transfer Learning for Image Classification," in *Advances in Computational Intelligence Systems: Contributions Presented at the 18th UK Workshop on Computational Intelligence*, 2018.