MIJST
MIST International Journal of Science and Technology

Intelligent Head-bot, towards the Development of an AI Based Cognitive Platform

Ramisha Fariha Baki[1], M. Akhtaruzzaman[2*], Tahsin Ahmed Refat[3], Mouneeta Rahman[4], Md Abdur Razzak[5], Md Mahfuzul Karim Majumder[6], Md Adnanul Islam[7], Md Meftahul Ferdaus[8], Muhammad Towfiqur Rahman[9], and Quadri Noorulhasan Naveed[10]

[1-6]Department of Computer Science and Engineering, Military Institute of Science and Technology (MIST), Dhaka, Bangladesh
[7]Department of Human-Centred Computing (HCC), Action Lab, Monash University (MU), Australia
[8]Department of Computer Science, University of New Orleans (UNO), USA
[9]Department of Computer Science and Engineering, University of Asia Pacific (UAP), Dhaka, Bangladesh
[10]College of Computer Science, King Khalid University, Abha 61413, Saudi Arabia

emails: *[1]ramishabaki@gmail.com; [*2]akhter900@gmail.com; [3]tahsin92refat@gmail.com; [4]mouneetar@gmail.com; [5]razzaksignals@gmail.com; [6]mahfuz25th@gmail.com; [7]adnan.islam@monash.edu; [8]mferdaus@uno.edu; and [9]towfiq@uap-bd.edu; and [10]qnaveed@kku.edu.sa*

ARTICLE INFO

Article History:

Received: 22nd July 2023
Revised: 02nd October 2023
Accepted: 22nd October 2023
Published: 28th December 2023

Keywords:

Cognitive humanoid head
Head-bot
Social robot
AI chat-bot
Machine intelligence

ABSTRACT

A cognitive humanoid head is an AI enabled head-bot platform that resembles human's cognitive abilities, such as perception, thinking, learning, and decision-making. The platform is able to interact with human through natural language processing and recognize individuals, thus allowing seamless communication between two parties. No such cognitive platform has been introduced in Bangladesh, thus creating an open field to contribute to the field of Machine Intelligence. This study aims to develop an AI based humanoid head (head-bot) capable of imitating a range of expressions, recognizing individuals, and interacts with visitors through general conversation. The head-bot skeleton is developed using a number of hexagonal blocks of PVC sheet to mimic a human-head-like structure where LCD, camera, microphones, and speaker are mounted. Two separate Machine Learning models are designed for face detection and recognitions, and voice enabled chat-bot implementation. The head-bot platform incorporates 2-DoF neck movements for various head gestures and face tracking. The Artificial Neural Network models are tested with accuracy of 95.05%, and 99.0%, for face detection and recognitions, and speech recognitions and response generation, respectively. According to the overall results and system performances, it seems that the proposed system has a number of good potentials for real life applications such as entertainment, guidance, conversations, interactive receptionists, personal companion, medical assistance, and so on.

1. INTRODUCTION

Humanoid robot development has become more agile, stable, and capable of all kinds of human-like abilities, including running, climbing stairs, carrying large loads, interacting with social environment, etc. (Sanaullah *et al.*, 2022; Bogue, 2020; Akhtaruzzaman *et al.*, 2017; Akhtaruzzaman & Shafie, 2010 a, b). The behavior of a humanoid robot can be directed into two research areas, task-oriented robots and interaction-oriented social robots (Akhtaruzzaman *et al.*, 2016; Kanda *et al.*, 2002). One of the main manifestations of human desire is to create an artificially intelligent social being as ultimate companion and assistant for human (Brooks, 1996). As such robots are expected to generate emotional engagements, designing a social robot-head is imperative (Breazeal, 2000). The fundamental and effective expressions can be displayed through humanoid head by moving neck, eyes, eyebrows, lips, and nose (Fong *et al.*, 2003), thus a robotic head with digital cognitive abilities is becoming increasingly important in Machine Intelligence (MI).

Humans respond better to robots that are empathetic, so a humanoid head with digital cognitive skills is introduced in this study aiming to bridge the gaps between social and artificial beings. In Bangladesh, it is rare to find a physical platform to implement theoretical knowledge and ideas of MI and Machine Learning (ML). The applied research in this field is very scattered and not integrated. Moreover, the applied fields of image recognition, voice recognition, Natural Language Processing (NLP), social interactions, and personal communications have not reached to their mature level, ensuring the opportunities for potential contributions.

The research aims to design and develop a cognitive humanoid head with basic functionalities and cognitive communication abilities. The understanding and modeling of elements within a variable context are also addressed in this study. The development of the system mainly followed four steps, first, designing a physical structure of a human head-like system having two degrees of freedom (2-DoF) at the neck, display unit, camera, and two microphones; second, designing a video processing unit for face detection and recognition with some basic capabilities of animated facial-like expressions through onboard display; third, designing the voice signal processing unit for speech recognition and response through speech; and finally, integrating all the individual modules and conduct overall system testing and performance analysis.

2. BACKGROUND STUDY

In the context of human-robot-interaction (HRI), a robotic head should be viewed as a sensory medium of communication. So, a robotic head must mimic a human head-like appearance in design where the concept of Golden ratio can be applied (Akhtaruzzaman & Shafie, 2011 a, b). The system requires to be designed as a mechanical structure that contains integrated sensing devices (Rojas-Quintero and Rodríguez-Liñán, 2021). A research conducted by Wang *et al.* (2014) proposed that the design of a human like robotic head must consider three things; avoiding the relationship between human beings and robots, understanding the association between a human face and a robot head, and constructing the robot heads through a series of experiments and testing. Reggia *et al.*, (2018) proposed a mechanism for learning by imitating a human where the robot used cause-and-effect reasoning to infer a demonstrator's goals in performing a task rather than just imitating the observed actions.

An empathetic robot must possess the ability to convey emotions through their facial expressions (Rawal & Stock-Homburg, 2022). Basically, a humanoid head-like system is a complex somatosensory structure which allows an android to act on its environment through its *roboception* (human-like perception) (Augello *et al.*, 2023). To investigate social interaction and human cognition, Cominelli *et al.* (2021) presented a cognitive humanoid robot with interactive head structure by integrating human-like emotions and time perceptions. The robot was used as a tool to study human mind, behavior, and consciousness.

Face detection is the area of image processing where the input picture is cropped and compared with the database to deduce the result of matching (Meena & Sharan, 2016). The general workflow of the face detection and recognition can be depicted as shown in Figure 1. Some previous studies developed the face recognition system based on geometric feature-based methods and template-based methods (Belhumeur *et al.*, 1997; Beymer, 1994; Turk & Pentland, 1991), but the recent researches mostly focus on model-based methods (Milad & Yurtkan, 2023; Dhamija & Dubey, 2022; Adjabi *et al.*, 2020; Nefian & Hayes, 1998). Some notable models applied in face recognition include Hidden Markov Model-based system, Eigenfaces using Principal Component Analysis, Fisher faces using Linear Discriminant Analysis, Bayesian method using Probabilistic Distance Metric, etc. The techniques are proven to be efficient in segmenting multiple human faces even with partial occlusion and complex cluttered background (Asha *et al.*, 2022; Huang *et al.*, 2022; Chen *et al.*, 2022; Iqbal *et al.*, 2011; Samaria & Young, 1994).

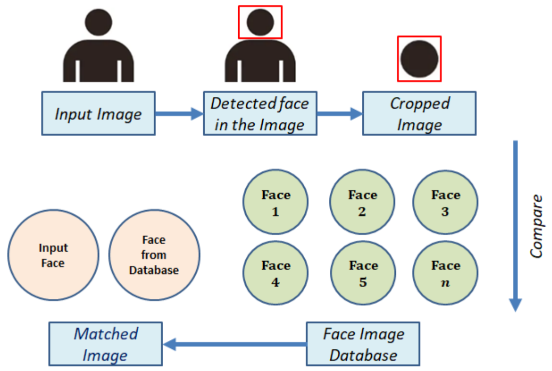


Figure 1: Conceptual model of face recognition and matching (Meena & Sharan, 2016)

Segmenting the faces in a video sequence requires a different approach as introduced by Li and Ngan (2008) which was based on Facial Saliency Map (FSM) for a head-and-shoulder type video application. The process consists of three stages, one: producing the saliency map of the input video image, two: localizing the face region in accordance with the saliency map using a geometric model where an eye map is constructed from chrominance components; and three: the adaptive boundary correction and the face contour extraction. The optical-flow technique is one of the most recent techniques for detecting human faces from a video. In this strategy the color information is extracted from the video frames to identify the possible location of the human face in the image. The method can also be applied for segmenting multiple faces (Sudha & Suganya, 2023; Iqbal *et al.*, 2011). The chat-bot technologies are used extensively nowadays for various purposes (Ahmed *et al.*, 2023; Apu *et al.*, 2022). An AI enabled head-bot must have the speech recognition capabilities where the automatic speech comprehension model can be designed through natural text parser as NLP for semantic representation, inference modeling, and selection of emotional behavior (reply speech and gestures) (Kotov *et al.*, 2023). The basic model of a humanoid chat-bot can be presented as shown in Figure 2.

Beside the powerful capabilities of the well-known AI-driven NLP tool, the Chat Generative Pre-trained Transformer (ChatGPT), several interesting researches are in full swing. Alekseev *et al.* (2021) presented a chat-bot based on ML model (accuracy, 97.0%) to automate the educational process where cross-platform Telegram messenger was used. In healthcare sectors AI chat-bots are also frequently being used with a good prospective where a transformer architecture based chat-bot model has a great potential if trained with a large amount of data (Li, 2023). A survey (with 408 respondents) on the application of chat-bot in small business for customer satisfaction is conducted recently by Rizomyliotis *et al.* (2022) showing very promising results of the embedded services in customer communication. Application of AI chat-bot also influence children to grow their learning interest is proven in another recent study conducted by Liu *et al.* (2022). Table 1 presents a short overview of some literatures in terms of the outcomes and limitations.

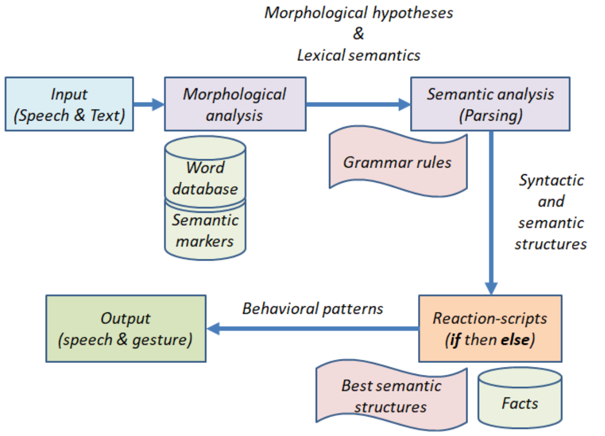


Figure 2: Conceptual process diagram of a humanoid chat-bot (Kotov *et al.*, 2023)

Table 1
Overview of some studied literatures

References	Objectives	Limitations
Kotov et al., 2023	A robotic platform with NLP based text parser and automatic speech comprehension	• Proposed approach covers below 50% of incoming utterances • Not a humanoid head-like chat-bot
Augello et al., 2023	Artificial somatosensory model of a cognitive humanoid	• A toy-robot like appearance • Not a general model, rather more specific to the used robot
Alekseev et al. (2021)	A ML model based chat-bot system through speech recognition and processing	• Absent of head-like chat system • A limited number of ML models are applied
Reggia et al., (2018)	Humanoid cognitive robot that learns to perform tasks via imitation learning	• Uses display unit for digital eye based expression • High computational overhead because of the application of neuro-cognitive control systems
Meena & Sharan, 2016	A face detection and recognition system	• High computation time for large image with high resolution • Not a human head-like chat-bot system

3. SYSTEM ANALYSIS AND DESIGN

The overall system is designed with six individual units with internal communication. The modules are, the humanoid head-like model with display, microphone, speaker, and camera; the video and image processing module connected with camera and Central Processing Unit (CPU); the audio processing unit connected with microphones and CPU; the eye-expression control unit connected with CPU and the head Liquid Crystal Display (LCD); the motor control unit connected with CPU through an Arduino board; and the CPU itself. The basic architectural diagram of the system is shown in Figure 3.

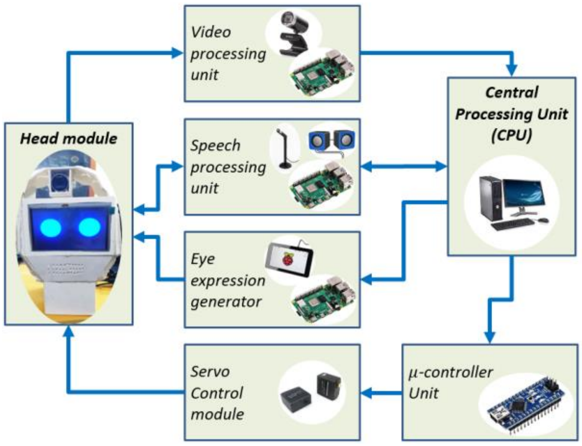


Figure 3: Basic system architecture block diagram

A. *Designing the Head-like Skeleton*

The head frame is developed using about 50 pieces of hexagonal blocks of PVC sheet (thickness, 2.0mm) attached together using hot-glue. Figure 4 shows the design of hexagonal block and the actual PVC blocks where d_c is the circumcircle diameter, d_i is the inner circle diameter, and x is the length of each side. The hexagonal shape is considered as the shape is the best fit to fill a curvy surface with no wasted space. The shape also minimizes the perimeter for a given area because of its 120° angles of each inside corner (Li *et al.*, 2014; Wang *et al.*, 2008). The hexagonal shape also possesses the beauty of the Golden ratio (ϕ) (Akhtaruzzaman *et al.*, 2022). Some larger blocks are used at the mouth and neck regions. The display is placed at the eye region, camera at the forehead region, two microphones at the ear regions, and the speaker at the mouth region (underneath the mouth block having smaller holes). Two motors are placed inside the neck region to provide necessary motions to the head-like structure.

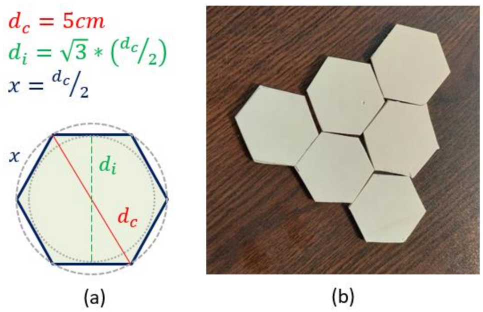


Figure 4: Design of hexagonal block for the development of head module; (a) Dimension of hexagonal block, and (b) Actual blocks of PVC material

The back side of the structure is eventually kept open so that all the associated items and electronics can be placed easily. Figure 5 presents the basic skeleton of the head-like interactive chat-bot system. The total weight of the head skeleton with associated components is calculated as $\approx 800\, gm$. Note that all the controllers and CPU are considered as off board. Table 2 shows the associated component specifications and total calculated weights.

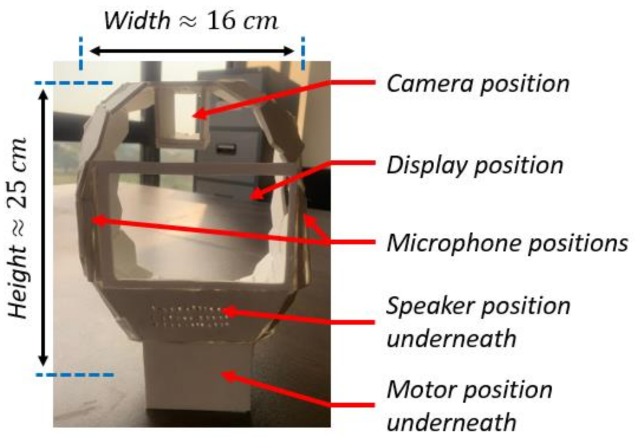


Figure 5: PVC sheet based skeleton of the humanoid head-like chat-bot prototype

Table 2
Head associated component specifications

Components	Specifications	Weight (gm)
PVC Sheet	Thickness: 2.0 mm	400
LCD Display	Dimension: ($13 \times 10 \times 3\ cm$)	175
Arduino Uno	Clock speed: 16 MHz Memory: 2KB SRAM, 32KB FLASH, 1KB EEPROM	30
USB Microphone	Frequency Response: $30Hz - 15kHz$ Impedance: $< 10\ K\Omega$	$(40 \times 2) = 80$
Webcam	Image sensor: 1080p Full-HD Frame Rate: 30 fps Still Image: 16 $Megapixel$	60
Speaker	Power input: DC 5V Impedance: 3Ω	10
	Total weight (m):	755 (≈ 800)

The calculated weight of the humanoid head is used for torque analysis which leads to select the appropriate motors with necessary rotational forces. The required torque for neck movements are calculated based on Equation (1) (Akhtaruzzaman & Shafie, 2011 May). In this case, the tilting (head nodding) motion angles of head is considered as not more than 20° from vertical axis. The required torque (τ) is calculated as $0.483\ N.m$.

$$\tau = m \times g \times l \times \sin(\theta) \tag{1}$$

Here, m is total mass ($m = 800\ gm$), g is gravitational force ($g = 9.81\ m.s^{-2}$), l is the distance of Center of Mass (CoM) from the rotational joint ($l = 18\ cm$), and θ is the considered tilting angle of the neck ($\theta = 20°$). Finally, the calculated required torque (τ) is multiplied with safety factor ($f_s = 1.5$) and final required torque, τ_f, is determined as ($\tau_f = (\tau * f_s) = 0.8\ N.m$). Now, the actuator is chosen as Hiwonder LDX-227 Servo Motor whose operating torque is $1.471\ N.m > \tau_f$. The neck motion ranges are demonstrated in Figure 6.

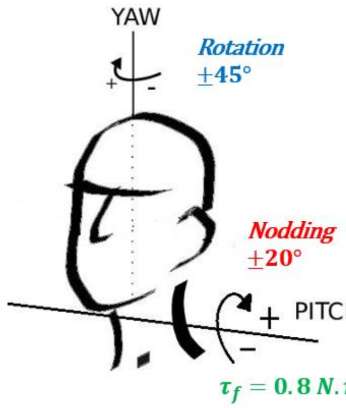


Figure 6: Neck motion ranges and required torque

B. Deep Learning Model for Face Recognition

Face recognition is the process of locating and identifying faces (mainly human faces) in a picture or a video sequence. Through identification of faces a person can be recognized and verified automatically where a computer vision based ML model uses pre-trained images of users' faces. FaceNet library powered by Google is used in this project to serve this purpose. Face embedded high-quality features can be extracted from face images using the FaceNet library to train a face identification model. The overall training process of the face recognition unit of the proposed chat-bot is shown in Figure 7.

To train the deep learning model, a dataset of about 6000 records are prepared and annotated. A consent form was designed for the student participants (more than 20) of the CSE Department of MIST to record their face images. About 80 to 100 images were taken for each person ensuring all possible angles of face view. The captured images of an individual were preprocessed and 200 to 250 images were generated for each participant for model training. The overall solution of the model is formulated in three stages; i) Pre-processing: which involves taking a collection of photos and converting them all to the same format, for example, square sized picture of a person's face; ii) Embedding: which is a crucial step FaceNet operation that ensures the representation of faces to learn in a multidimensional space where distance is a measure of face similarity; and iii) Classification: which uses data from the embedding process to distinguish between different faces.

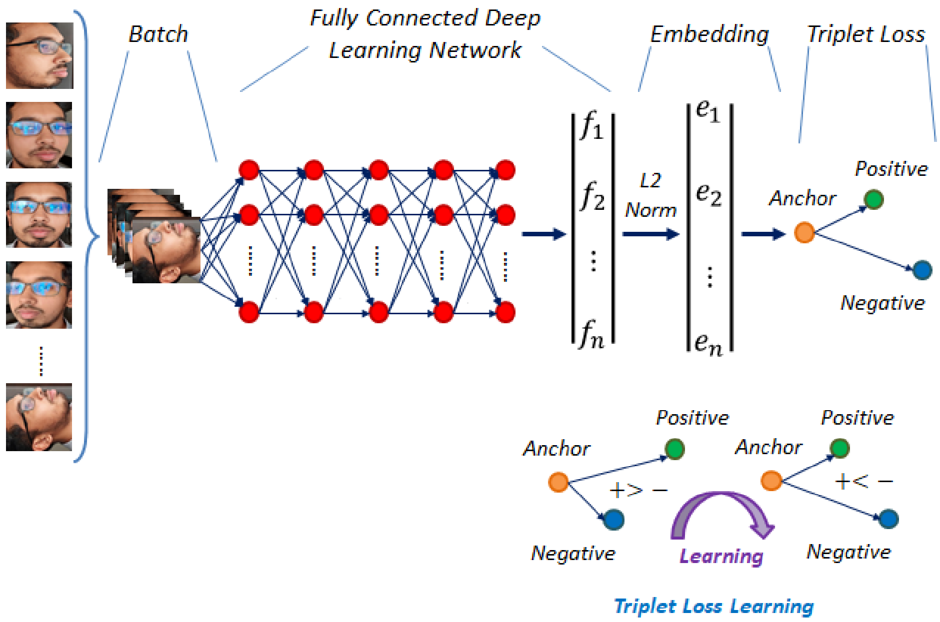


Figure 7: Overall architecture of the fully connected deep learning model for face recognition

C. Speech Recognition and Feedback Processing

This module consists of a microphone (for voice input), a speaker (for voice output), and a CPU (for processing and generating replies). As the functionality of the chat-bot head is designed based on collected data focusing the CSE department of MIST, the results mainly supports the relevant queries. To design the speech support model, five algorithms (Naive Bayes, Logistic regression, K-neighbors classifier, Decision tree, and Artificial Neural Network (ANN)) are applied to train the system separately based on the same dataset. The ensemble learning strategy suggested the ANN model because of the highest accuracy over the other algorithms.

The ANN model is designed based on three-layer architecture consisting an input layer, 8 hidden layers, and a output layer. The steps of the overall process are; i) Preprocessing: the queries of the dataset need to be pre-processed before feeding into the ANN model where the $nltk(NLP)$ library is used; ii) Model training: the preprocessed data is feed into the ANN model which is designed based on the neural network library of $PyTorch$ using ReLU activation function; and iii) Deploying the trained model: the trained model is used to predict the tag of a query and later a separate script selects a relevant response randomly from the corresponding response set and converted into speech signal. The overall flow diagram of the module is demonstrated in Figure 8.

The dataset for the speech processing unit is generated based on the possible queries about the department of CSE of MIST, their relevant answers, and corresponding tags. In total of 800 different queries and 1000 distinct answers under about 200 tags are recorded to generate the dataset. The format of the dataset is a JavaScript Object Notation (JSON) script having several lists of strings (queries and responses) and individual tags. A tag, a query list, and a response list make an object, where several such objects create an Intents list under the master object. The structure of the python script is shown in Figure 9. The basic architecture of ANN model for the speech module is illustrated in Figure 10.

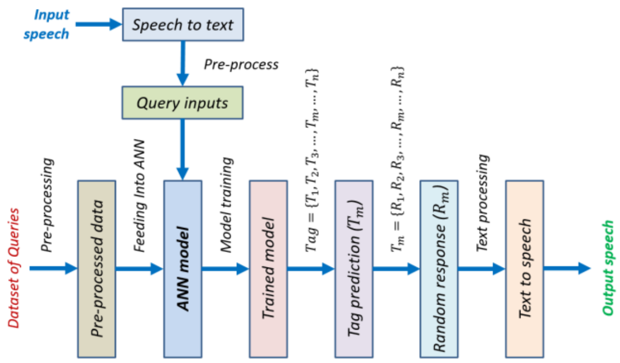


Figure 8: Workflow of the speech processing module

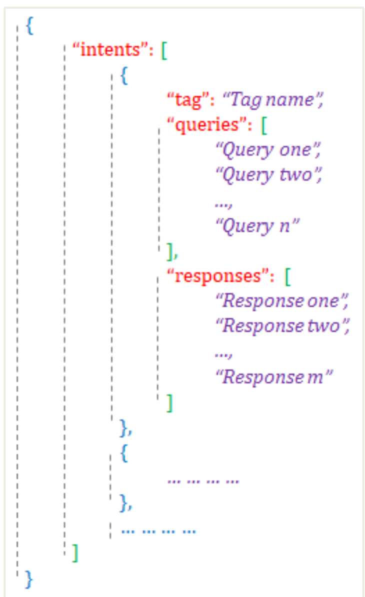


Figure 9: JSON structure for speech processing dataset

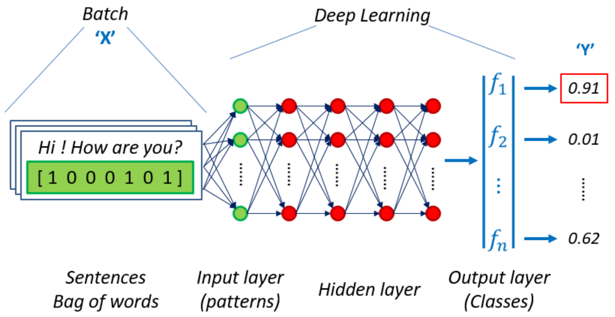


Figure 10: ANN model for speech processing

D. Design of Animated Eye-expressions

Four different eye-expressions (happy, sad, angry, and normal) are designed to animate on the display unit reflecting the visual feedback to a situation and user activity. Happy eye-expression is activated if a visitor's face is recognized, a greeting type of answer is delivered, or saying goodbye to a user. The sad eye-expression appears if the person in front of the robot-head is not recognized; the head-bot is unable to provide any exact answer; the question (any speech) is not understandable (not a valid processing result) to the head-bot, for example more than one person asking questions at the same time; or there is a very loud noise in its environment. The angry (or disgust) eye-expression is animated on the display if anyone uses any abusive words, any query remains incomplete, or the robot asks to repeat the question. If the head-bot is idle or have no suitable inputs, normal eye-expression is animated on the display. This expression also appears for some other situations like continuous conversation, pauses during conversation, or interacting with an unknown user. The basic design of the four expressions is shown in Figure11.

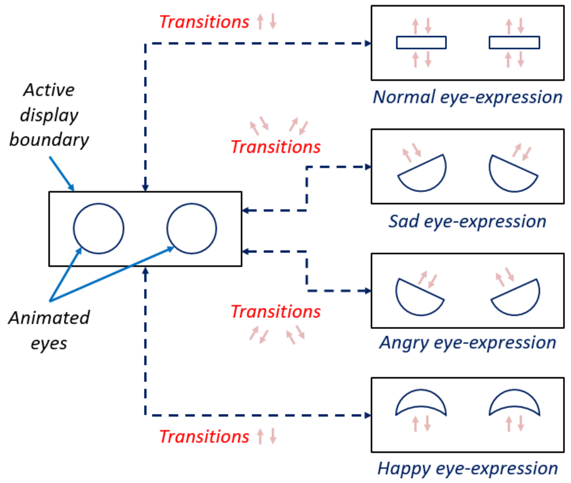


Figure 11: Basic design of four types of eye-expressions

E. Design of Face Tracking Control Algorithm

The servo motors are connected with the Arduino Nano μ-controller board where a virtual feedback-control algorithm (likely a proportional-derivative (PD) control architecture) is implemented. The purpose of the design of the PD-control algorithm is to provide a smooth transition from initial angular position of the head-bot to its desired angular position. The smooth transitions of positions will enable the system to present its demeanor in a realistic way. Figure 12 presents the control architecture of a single actuator for smooth transitions of head-bot motions.

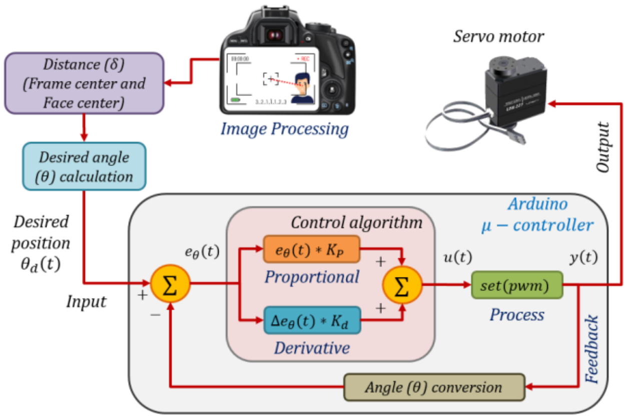


Figure 12: Face tracking control engineering

A python script determines the distance between image frame-center and the recognized face-center. The distance is decoded to the desired angular position (θ_d) of the actuator. This θ_d is the input of the control architecture which is summed up with the negative feedback (present angular position) of the control mechanism. The current angular position is mapped based on the PWM vs. angle table as described in the datasheet of the actuator. At this stage the calculated error ($e_\theta(t)$) is used to determine the proportional and derivative control signals where the control gains, K_P and K_d are selected as 0.04 (4.0%) and 0.96 (96.0%), respectively, through several trial and errors. The both values together produce a value as 1.0 (100%), thus the summation of both the control signals ($u(t)$) does not overshot the initial and desired angular position boundaries. Finally, the produced control signal ($u(t)$) is fed to the position to PWM conversion process to produce the ultimate output signal ($y(t)$) and directly feed to the actuator. The control signal, $u(t)$, calculation process can be described through Equation (2). The ultimate schematic diagram of a servo control circuit is shown in Figure 13.

$$\left.\begin{aligned} &\mathrm{u(t)} = \big((e_\theta(t) \times K_P) + (\Delta e_\theta(t) \times K_d)\big) \\ &K_P = 0.04 \;\&\; K_d = 0.96 \end{aligned}\right\} \qquad (2)$$

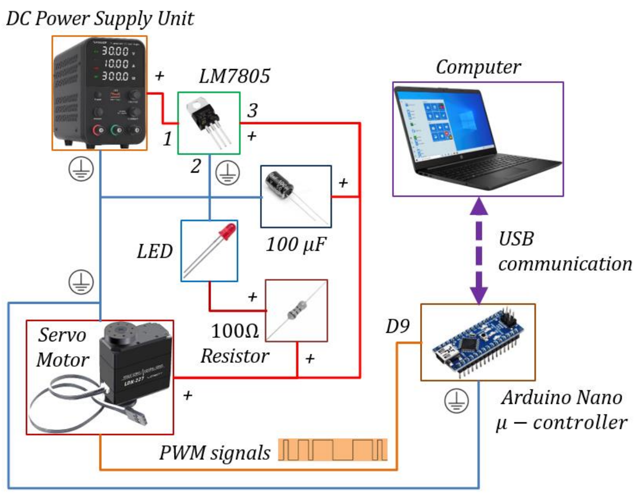


Figure 13: Schematic diagram of servo control circuit

4. MODELS IMPLEMENTATION AND TRAIN

A. Face Detection and Recognition

The gathered images are augmented to increase the content of the dataset for model training and testing. The Python's AUGMENTOR package is used for image enhancements and augmentations. Samples of some collected images are depicted in Figure 14.

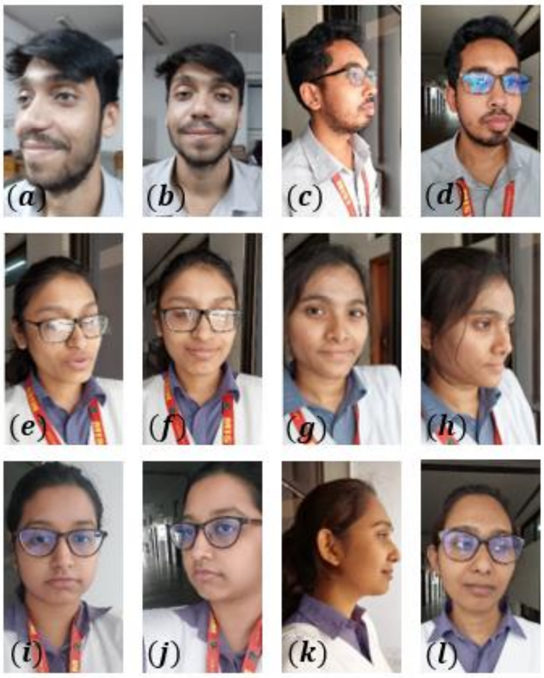


Figure 14: Sample images of the collected data, (a) & (b) Sample-A, (c) & (d) Sample-B, (e) & (f) Sample-C, (g) & (h) Sample-D, (i) & (j) Sample-E, (k) & (l) Sample-F

A big dataset is challenging to produce but affects face recognition accuracy. In this case, image augmentation is imperative which increase the number of images resulting increase of the volume of the dataset. The collected images are altered to produce new images applying a variety of transformation, such as rotating, flipping, shifting, sharpening, brightening, fading, darkening, etc. The AUGMENTOR python library is customized to do the tasks easier for image enhancement and creating the artificial image data for ML model training. The python scripts also have some image pre-processing features that provide extra facilities for image modifications. Sample of augmented pictures are given in Figure 15.

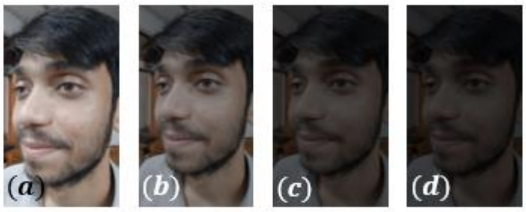


Figure 15: Sample images of various augmentation stages, (a) original image, (b) first step augmentation, (c) second step augmentation, and (d) third step augmentation

The facial recognition system FaceNet, developed by Google, uses a Deep Convolution Neural Network (DCNN) that maps the images of a person's face into Euclidean spaces (collections of geometrical points). The process is called image embedding where the values are obtained based on the level of similarity and differences of faces. On the other hand, face detection indicates finding faces in photos automatically and localizing them by enclosing with a bounding box. In this study, only the aligned images are taken (segmented) to prepare the dataset. The face alignment is conducted based on the bounding box of the face, from which only the facial features are extracted. Examples of segmented faces are presented in Figure 16. Some of the sample face images used for model training are shown in Figure 17.

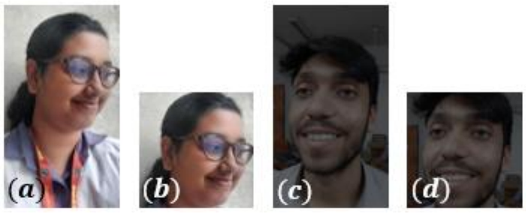


Figure 16: Image before and after segmentation, alignment, and cropping, (a) & (b) Original and segmented images, (c) & (d) Augmented and segmented images

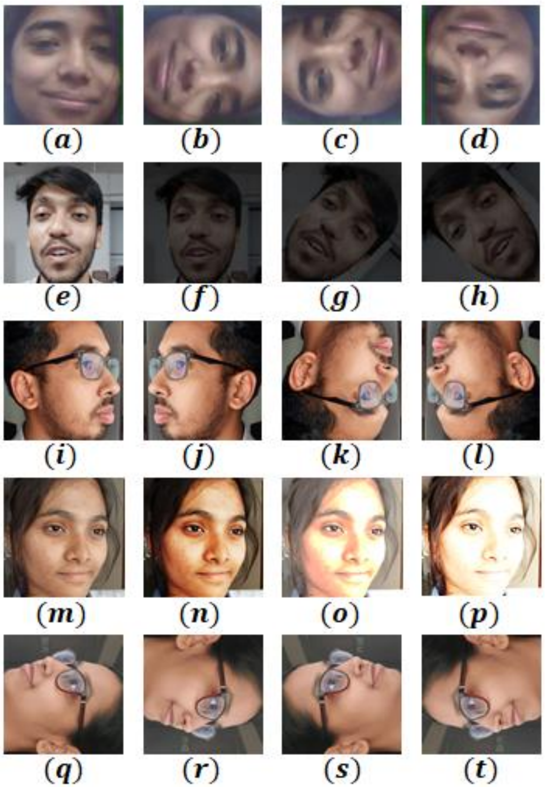


Figure 17: Augmentation of segmented images; (a) actual segmented image of a sample; (b) $+90°$ rotation of image (a); (c) $-90°$ rotation of image (a); (d) 180° rotation of image (a); (e) segmented image of a sample; (f) darken image of (e); (g) $+45°$ rotation of image (f); (h) $-45°$ rotation of image (f); (i) segmented image of a sample; (j) horizontal flip of image (i); (k) vertical flip of (i); (l) vertical flip of (j); (m) segmented image of a sample; (n) sharpen image of (m); (o) 40% bright of image (m); (p) 60% bright of image (m); (q) $+90°$ rotation of a sample image; (r) vertical flip of (q); (s) horizontal flip of image (q); and (t) vertical flip of image (s)

Preprocessing of an image suppresses the unnecessary distortions, improves image information, and enhances some important features used for further processing. Geometric transformations of images (like rotation, scaling, translation, etc.) are also involve in pre-processing. In the facial recognition context, the process also solves a few problems like lighting differences, occlusions, alignments, and segmentations. The segmentation problem is solved by finding the largest face in an image. In a photograph, it is common for a face not to be aligned perfectly within the frame. This problem is solved based on the position of the eyes and bottom lip and the photos are translated to match with the center of the bounding box and to standardize the cropped face. Facial recognition is a biometric solution that measures the unique characteristics of one's face. FaceNet relies on the image pixels as features, rather than extracting them manually. For the loss function, FaceNet uses "triplet loss" that relies on minimizing the distance from positive examples while maximizing the distance from negative examples. Faces that belong to same identity (cluster) need to be close (short distance) from one another than the ones belonging to another cluster (identity).

B. Chat-bot Implementation and Training

For the chat-bot (of the head-bot) implementation and training, the collected data is based on the information about CSE department of MIST. The data consists of the queries that may arise during the visit at the department by the guest visitors, students, and other staffs. Also, the possible and appropriate responses to those queries are pre-determined and stored in the designed structure. The collected data are preprocessed as described below.

- Firstly, tokenization is conducted by converting a sentence into a list of corresponding words. For example, the query "*Is anyone there*?" is converted into the list as ["*Is*", "*there*", "*anyone*", "?"].
- Second, all the uppercase letters are changed into lower cases with stemming. This process converts the above list into ["*is*", "*there*", "*anyone*", "?"]. Stemming defines to find root words with similar origin. This is required to remove the redundant words from the prepared tokens.
- Third, all the punctuations are removed. At this stage the list becomes ["*is*", "*there*", "*anyone*"].
- Fourth, the bag-of-words are generated by processing the existing lists of a particular tag of queries. Here the binary representation of the dataset is generated and compared with the query-list to produce the bag of words. For instance, the binary representation of the above list of a particular tag becomes $[0,0,0,0,0,0,0,0,1,1,1]$ if the actual list of words for that particular tag is ["*hi*", "*hello*", "*how*", "*are*", "*you*", "*bye*", "*see*", "*later*", "*is*", "*there*", "*anyone*"]. This binary list is basically the input layer of the ANN model for speech processing.

The ANN is the foundation of the deep learning (the subset of ML) that reflects the structure and function of the human brain (Bishop, 1994). The input data (binary list) is an individual query where the appropriate tag for that query is predicted by the trained model. Finally, a random response (from the response list) under that predicted tag is considered as output. This output with the response is converted into speech signal using a python script and sent to the speaker module of the head-bot. The overall workflow of the audio and speech processing module is described pictorially in Figure 18.

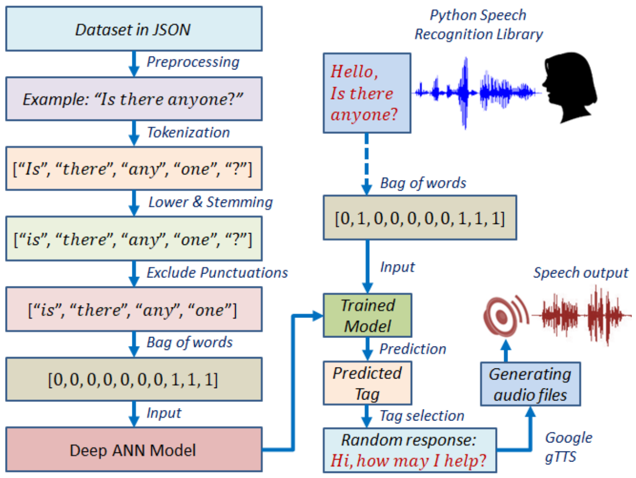


Figure 18: Overall workflow of speech recognition and output processing

C. Implementing Eye-expressions

The Blender software is used to design the eye expressions in animated "*.avi" video file format. Firstly, using the software, a plane is designed and textured to make the display panel. Second, two circles are designed with shaded blue color to represent the eyes. In this case bloom option is used to adjust the brightness and to create a laser-like effect. After that, key-frames are added in four planes for rendering so that the eyes have various expressions. Two types of panels, rectangular and circular, are created to express the emotions. The two rectangular panels are placed at the upper and lower portions of each eye for sliding over the eyes generating sad, angry, and normal expressions. Small angular rotations for the rectangular panels are generated only for the sad and angry expressions. The circular panels slide over the eyes from the bottom (slightly angular directions) creating the happy expression. The designed eyes and the four planes are demonstrated in Figure 19. A Python script is designed to run and stop the video sequences for each eye-expression based on the various commands generated by the head-bot control architecture. The overall process flow diagram of eye expressions control is presented in Figure 20.

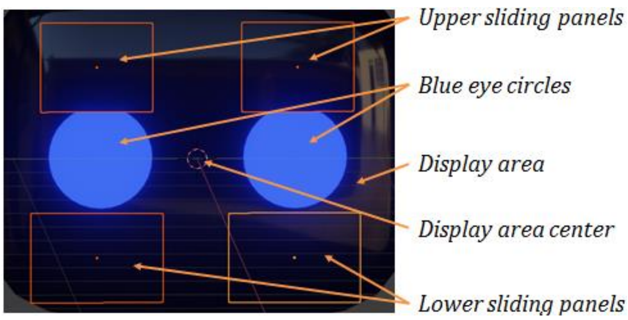


Figure 19: Design of eye-expression animations using Blender software

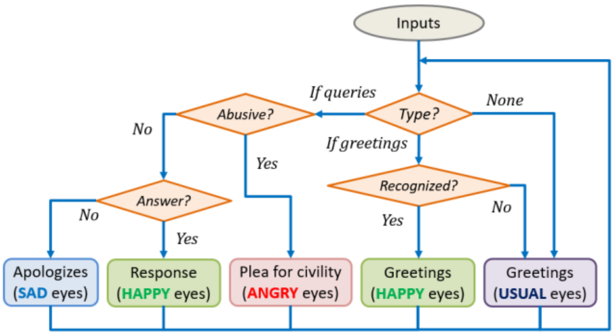
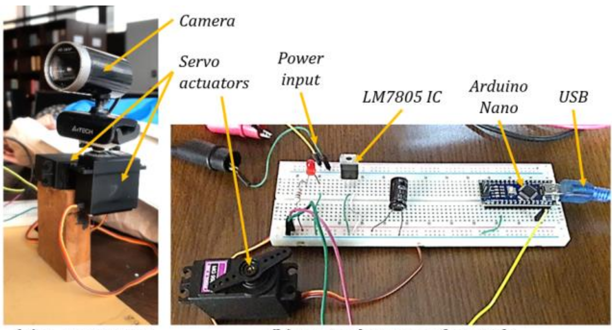


Figure 20: Eye-expression control architecture

For any input (mainly voice), the system checks the input types whether it is query or greeting. If none, the system maintains the normal (usual) eye-expression. For greeting, system checks the person is recognized or not. If yes, system animates happy expression with greetings, otherwise shows normal expression with greetings. If the input is a query, system checks if there are any abusive words. If so, the head-bot presents angry eyes with request not to use any abusive words. If not, then the chat-bot checks for valid answers, if available then it response with happy eye-expression, otherwise apologizes with sad eyes.

D. Implementing 2-DoF Neck Servo Control

The designed virtual PD control architecture for head-bot neck movements for face tracking is implemented in the Arduino Nano μ-controller board. The experimental setup for 2-DoF joint with two servo motors having the mounted camera is presented in Figure 21(a). The setup is mimicking the head-bot neck motion in terms of pitch and yow movements. Figure 21(b) presents the basic circuit setup for single servo actuator control.



(a) 2-DoF active joint with camera *(b) Control circuit of a single servo actuator*

Figure 21: Experimental setup for testing: a 2-DoF active joint for pitch and yaw motion control of mounted camera

The feedback from camera module is processed in main CPU to determine the distance between the center points of image frame and detected face. Considering the distance as hypotenuse of a right-angle triangle, the Pythagorean formula is applied to determine the base (adjacent) and the perpendicular (opposite) of the triangle which leads to calculate the $\angle\alpha$ and the $\angle\beta$ as desired angles of pitch and yaw actuators, respectively. A pictorial description of the process is presented in Figure 22 where the line $A'C'$ is indicating the distance between the image-frame center and the face-frame center. The line forms a right triangle as projected in ΔABC. The $\angle\alpha$ and $\angle\beta$ are representing the projection of desired Pitch and Yaw angles of the actuators of the neck joint. The angles are calibrated with the actual rotations of the actuators through several experimentations. Necessary formulations to determine the angles are presented in Equation (3). Figure 23 demonstrates the overall logic architecture of head-bot motion control.

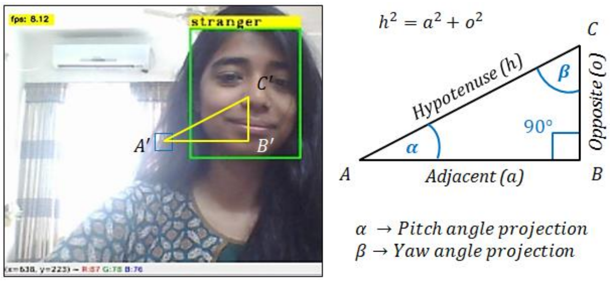


Figure 22: Calculation of pitch and yaw angle from distance geometry for face tracking

$$\left.\begin{aligned} \alpha &= sin^{-1}\left({}^{o}/_{h}\right) = sin^{-1}\left(\frac{\sqrt{h^2-a^2}}{h}\right) \\ \beta &= sin^{-1}\left({}^{a}/_{h}\right) = sin^{-1}\left(\frac{\sqrt{h^2-o^2}}{h}\right) \end{aligned}\right\} \qquad (3)$$

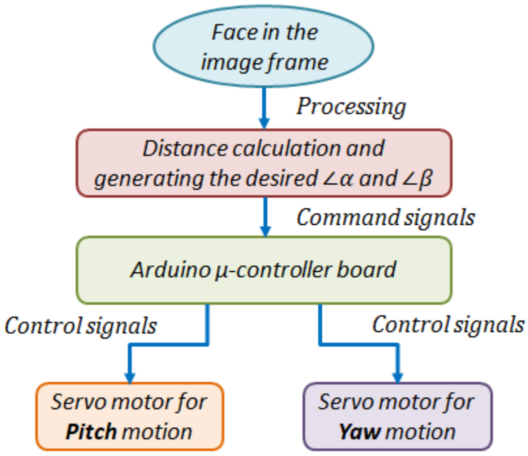


Figure 23: Overall architecture of head-bot motion control

E. The Integrated Module

Each of the individual modules is integrated together to form the final head-bot system. The camera, speaker, two microphones, LCD, and actuators are mounted on the head-bot skeleton with the necessary circuit, controllers, and CPU connections. The complete head-bot system is shown in Figure 24.

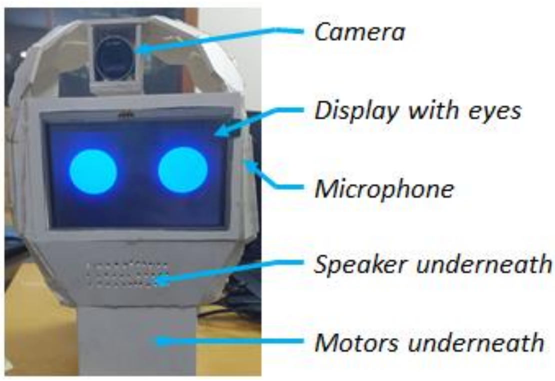


Figure 24: Integrated head-bot system

All the relevant coding with the four programming modules are combined into one process through Python multithreading library. To achieve the goals of the head-bot system, the individual thread needs to communicate one another to ensure the exchange of necessary data and information. In this case, some global variables are used so that they can be accessed at any time, given that the variable is not currently being accessed by other programs. The general program flow diagram of the head-bot system is shown in Figure 25.

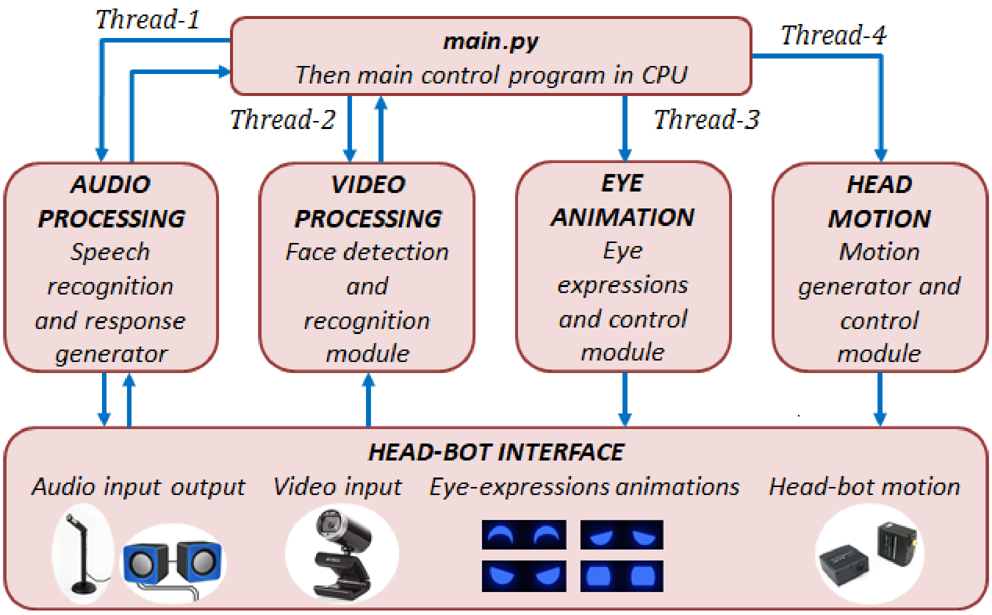


Figure 25: Overall program flow diagram of the head-bot system

5. RESULTS AND DISCUSSIONS

A. Face Detection and Recognition Module

Accuracy is one of the most commonly used measures for evaluating an ML model that reflects the effective prediction of specific outcomes relative to the actual truth. A model accuracy is determined by dividing the total number of true predictions by the total number of predictions, as presented in Equation (4). Greater accuracy of a model means the model will produce fewer false predictions, which is a positive indicator of the model.

$$Accuracy = \frac{Number\ of\ True{-}positive\ predictions}{Total\ number\ of\ predictions} \tag{4}$$

The designed model is trained with a custom made dataset of 6000 records of facial images. In the terminology of the triplet loss, the focus is always on the anchor image and the distance score between the anchor and the positive image (another image of the anchor) needs to be similar while the distance score between the anchor and the negative image (images other than the anchor) is comparatively higher. The similarity score is calculated by comparing the distance obtained from the triplet loss and the threshold.

The FaceNet library shows high efficiency and performance as 99.63% of accuracy for the Labeled Faces in the Wild Home (LFW) dataset and 95.12% of accuracy for the YouTube Faces Database while using only 128 bytes per face. However, based on the custom dataset of this study, the accuracy (for 80% - 20% train-test split) is determined as 95.05%. Corresponding results of Epoch vs. Loss and Epoch vs. Accuracy of the model training are presented in Figure 26. For the model training, ten epochs are considered while 600 records of images make a batch for each epoch. Only for this condition, the model that is presented in this article has achieved the best score.

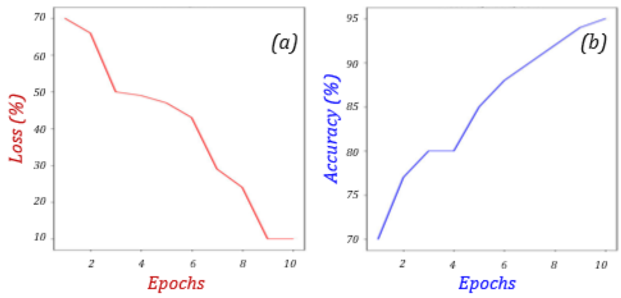


Figure 26: Model training performance graph, (a) Epoch vs. Loss, and (b) Epoch vs. Accuracy

This study considers a threshold of 87.0% as similarity score. Thus if the similarity of a test image (face) is more than 87% (threshold), it is considered that the person is known (recognized) thus the head-bot responses by mentioning the person's name. Each time a person faces the head-bot, the similarity level is generated. If the value become greater than the threshold the person is recognized. Otherwise, the person is considered as a stranger. A test result is presented in Figure 27 where (a) is the anchor image, (b) is a different image (with eye glasses) of the anchor, and (c) is an image of a different person. The designed model predicts the similarity score as ≈ 93.0%

for image (b) thus marked as positive, and the score as $\approx$ 25.0% for image (c) marked as negative.

(a) Anchor *(b) Positive [Similarity score ≈ 93%]* *(c) Negative [Similarity score ≈ 25%]*

Figure 27: Test results with random images based on the custom-made dataset

The designed head-bot's face recognition module is also tested for real time video sequence. One of the test results is demonstrated in Figure 28 where the system is detecting faces as well as recognizing if similarities are found in its trained model. Figure 28(a) is recognized face thus indicated by the person's name "$mouneeta$", 28(b) is recognized face having no attention, thus indicated by "?" mark, and 28(c) is the unrecognized face marked as "$stranger$" as no match found in the trained model.

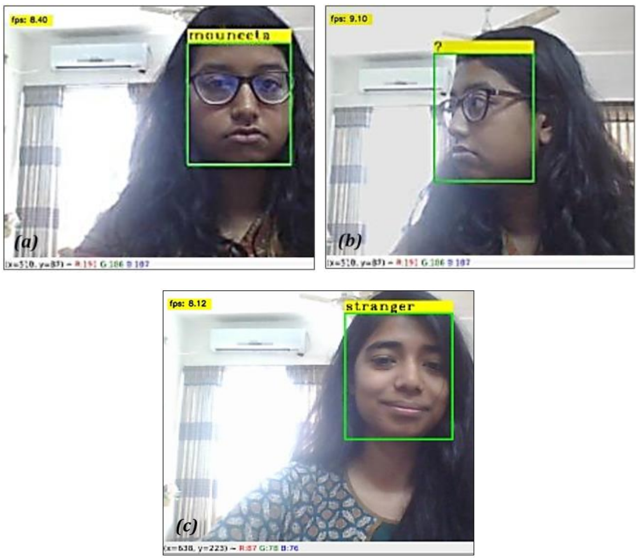


Figure 28: Results of face detection and recognition module; (a) person is recognized, (b) The face is not in focus though it is recognized, and (c) the face is not recognized and marked as "$stranger$"

B. Speech and Response Processing Module

The strategies of NLP are applied with the aid of ML techniques to design a voice based chat-bot for the humanoid head-like system. A custom created dataset is utilized to train the system where five different ML algorithms are applied to determine the efficiency and optimality by comparing the accuracy scores. The models are developed using python Natural Language Toolkit (NLTK) and tested using Pandas. Decision Tree (DT), Naive Bayes (NB), Logistic Regression (LR), K-Neighbors Classifier (K-NC), and ANN are applied to conduct the performance analysis. Though the accuracy is the most important measures in selecting the best strategy, in actual cases, highest accuracy does not delineate the concrete picture of a model performance; instead, additional metrics like precision, recall, and F1-score also need to be observed and analyzed. The necessary formulas are presented in Equation (5), (6), and (7). Here, TP is $true - positive$ and FN is $false - negative$.

$$Precission = \frac{Number\ of\ TP\ predictions}{Total\ number\ of\ positive\ predictions} \tag{5}$$

$$Recall = \frac{Number\ of\ TP\ predictions}{Number\ of\ TP\ \&\ FN\ predictions} \tag{6}$$

$$F1_{Score} = \frac{2 \times precission \times Recall}{precission + Recall} \tag{7}$$

The performance matrices of various algorithms are presented in Table 3 where the highest accuracy is observed for DT as 92.0%. Basically, for all the performance measures, DT shows the highest values for all cases. The second highest values can be found for LR algorithm. The NB performances are observed as very low comparing to other three strategies. The accuracy comparison among all four algorithms and the ANN model is shown in Table 4 where ANN model accuracy is the highest as 99.0%. The loss of the model is deduced as $\approx$ 0.0% after the 400^{th} epoch out of all 1500 epochs.

Table 3
Classification results of various ML algorithms

Measures	LR	K-NC	DT	NB
Accuracy	84%	79%	92%	70%
F1 score	82%	76%	90%	65%
Recall	84%	79%	92%	70%
Precision	85%	78%	90%	68%

Different ML algorithms perform differently because of their different design and nature. The DT algorithm is more suitable for classifications, thus showing better performances. On the other hand, complex models like ANN are capable of considering intricate patterns of data, thus display the best results. The training parameters of the chat-bot model is highlighted in Table 5.

Table 4
Accuracy comparison among five strategies

Algorithms	Accuracy
Logistic Regression (LR)	84%
Naive Bayes (NB)	70%
K-Neighbors Classifier (K-NC)	79%
Decision Tree (DT)	92%
Artificial Neural Network (ANN)	99%

Table 5
Chat-bot model training parameters

Number of Tags	60
Name of the Tags	'13', '13names', 'AEseatavail', 'ALevel', 'ARCHseatavail', 'AUnit', 'AUnitCost', 'AdminOffice', 'BMEapply', 'BMEseatavail', 'BUnit', 'BUnitCost', 'BatchDegreeYear', 'CEseatavail', 'CSE', 'CSEseatavail', 'Cafeteria', 'ChairmanAcademic', 'ChairmanCouncil', 'ChairmanGoverning', 'Clubs', 'Commandant', 'CyberRangeLoc', 'EECEseatavail', 'EWCEseatavail', 'G2 Sir', 'HOD', 'HOD office location', 'HSC', 'Hall', 'HeadCSE', 'IPEseatavail', 'Library', 'Location', 'MEseatavail', 'MIST', 'MISTTestablished', 'Medical', 'NAMEseatavail', 'NSEseatavail', 'OLevel', 'PMEseatavail', 'PostGradCourse', 'SSC', 'Scholarship', 'Sports', 'Stationery', 'Stipend', 'UnderGradCourse', 'Units', 'applicantsBsc', 'applyApplication', 'goodbye', 'greeting', 'nowlocation', 'qualificationLocal', 'seatsavailBsc', 'thanks', 'whoami', 'whoisbot'
Number of Epochs Used to Train	1500

C. Eye Expression Module

The eye-expression animated in the LCD is integrated with the face detection and the speech processing modules. For different scenarios of the two modules, different eye-expressions are selected and displayed. For each of the expressions, the control architecture chooses an individual video file (".avi" format of 500 ms video) and plays on the LCD screen. Results of the various eye-expressions are shown in Figure 29.

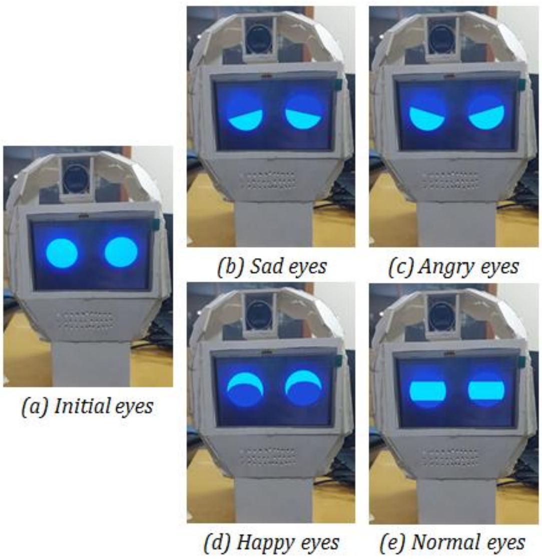


Figure 29: Results of the various eye-expressions

D. Neck Motion for Face Tracking

The designed face detection and recognition algorithms detect any face in the scope of the camera frame and draws a bounding box around the face. The algorithm locates the center of the frame and always tries to align the frame-center point with the center point of the face rectangle. The camera (head-bot) always moves to match the center points if the detected face moves within the image-frame (camera scope). The 2-DoF joint actuators drive the head-bot fast enough with the smooth pattern of motion trajectory because of the implemented PD-control algorithm. Four different test results of a single actuator control within 0° to 180° motion range are illustrated in Figure 30. On the other hand, three test results of face tracking (for three test subjects) by the prototype of the head-bot system is presented in Figure 31.

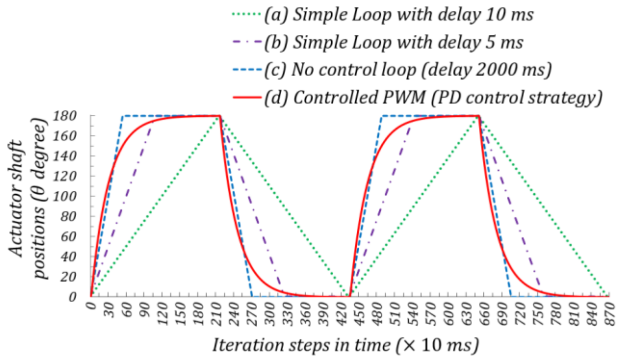


Figure 30: Results of the various eye-expressions

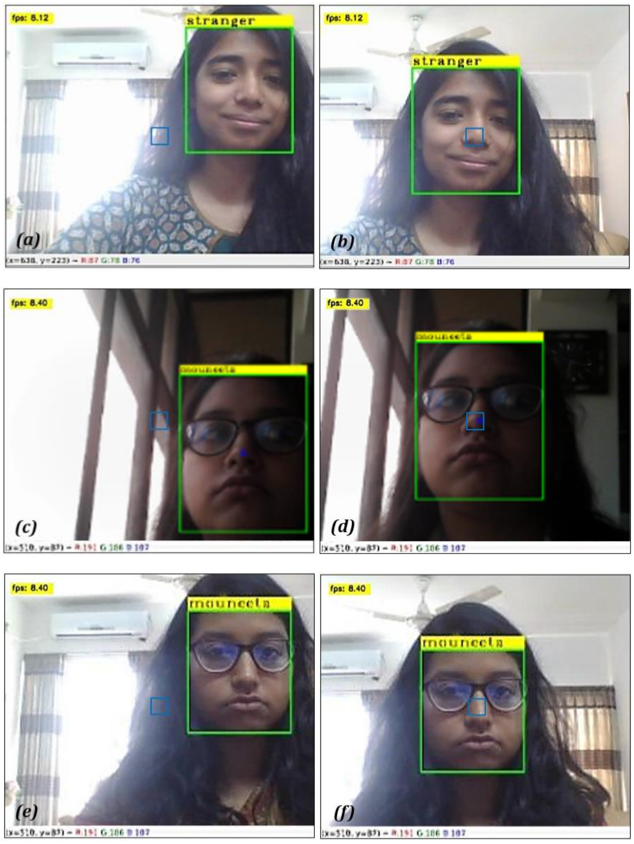


Figure 31: Three test results of face tracking by the designed head-bot system; before and after of the tracking operations, (a) & (b) for subject 1, (c) & (d) for subject 2, and (e) & (f) for subject 3

According to Figure 30, the graphs (a) and (b) presents the results of simple loop-based control strategies showing sharp transitions while reaching to the desired angular positions, $0°$ to $180°$ and vice versa. The similar results with very sharp transitions are also observed for loop less position transition strategies with one or two seconds ($1000\ ms$ or $2000\ ms$) delay. This sharp transition of position expresses the unfriendly behavior to the audiences. Moreover, the instant stop of a particular motion is harmful for the mechanical structure of the designed system as it creates jerk. The problem is solved by designing the PD-control algorithm which generates faster but smooth control of motion trajectory. The test results of the PD-control mechanism are presented in Figure 30(d) for the position transition from $0°$ to $180°$ and vice versa.

5. CONCLUSIONS

The head-bot system, presented in this study, mainly has four functionalities, the face detection and recognition functions, voice based chat-bot mechanisms, conveying emotions through animated eye-expressions, and head-bot motion control mechanism for face tracking. Though, several ML algorithms are tested to determine the best performances of individual modules, mainly ANN models are designed for final implementations. Results show that the ANN models for head-bot vision system produces 95.05% while the chat-bot module produces 99.0% accuracies. The results ensure that the individual modules are able to perform their jobs efficiently, thus overall system confirms its viability for real time applications, such as personal assistant, companion robot, chat-bot system, help and support, in the field of pedagogy and andragogy, medical support and help, entertainment, business receptionist, guide, and so on.

At this stage, the system has several limitations but has very high potentials for future research. The current design of the head-bot skeleton needs to be designed professionally to create the appearance similar to a human face. This appearance will increase the public acceptance of the system as a new platform of man-machine interactions, in the context of Bangladesh. Soft robot technologies can be applied to design and develop of the head-bot architecture.

At this moment the system response time is a little bit slow, thus creates an opportunity to improve the overall performances. The microphones used in this project are not well-suited to noise-canceling techniques. A reinforcement learning mechanism is not applied which necessary to make the system more intelligent is providing continuous learning abilities from its surroundings. Head nodding and shaking behavior may increase its gravity to the next level. Multiple language support can be introduced to make the head-bot more accessible.

The development of the cognitive humanoid head represents a significant leap forward in the field of AI. With the ability to perceive, reason, learn, and make decisions, this technology has the potential to revolutionize the way humans interact with machines. Though, the robotic-head presented in this study is still in its very early stage, the potential applications of such a system are immense. It will be exciting to see it continual evolution in the future.

ACKNOWLEDGEMENTS

Authors would like to express their appreciation to the Ministry of Defence (MoD), Bangladesh; the Department of Computer Science and Engineering (CSE), Military Institute of Science and Technology (MIST), Dhaka, Bangladesh for providing the available facilities and support during the study. The authors are also thankful to the research and development (R&D) team of DynREAM Robotics Ltd., Bangladesh, and the research team of Quantum Robotics & Automation Research Group (QRARG), Department of CSE, MIST, Bangladesh.

REFERENCES

Adjabi, I., Ouahabi, A., Benzaoui, A., & Taleb-Ahmed, A. (2020). Past, present, and future of face recognition: A review. Electronics, 9(8), 1188.

Ahmed, L., Polok, I. K., Islam, M. A., Akhtaruzzaman, M., Mukta, M. S. H., & Rahman, M. M. (2023, January). Context based Emotion Recognition from Bengali Text using Transformers. In 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 1478-1484). IEEE.

Akhtaruzzaman, M., & Shafie, A. A. (2010a). Evolution of humanoid robots and the contribution of various countries in advancing the research and development of the platform. In ICCAS 2010 (pp. 1021-1028). IEEE.

Akhtaruzzaman, M., & Shafie, A. A. (2010b). Advancement of android and contribution of various countries in the research and development of the humanoid platform. International Journal of Robotics and Automation (IJRA), 1(2), 43-57.

Akhtaruzzaman, M., & Shafie, A. A. (2011a). Geometrical substantiation of Phi, the golden ratio and the baroque of nature, architecture, design and engineering. International Journal of Arts, 1(1), 1-22.

Akhtaruzzaman, M., & Shafie, A. A. (2011b). An attempt to develop a biped intelligent machine BIM-UIA. In 2011 4th International Conference on Mechatronics (ICOM) (pp. 1-7). IEEE.

Akhtaruzzaman, M., Shafie, A. A., Raihan, S. M., Hasan, M. K., Ahsan, T., Alam, M. S., & Haider, M. B. (2011, December). Golden ratio, the Phi, and its geometrical substantiation. In 2011 IEEE Student Conference on Research and Development (pp. 425-430). IEEE.

Akhtaruzzaman, M. D., Shafie, A. A., & Khan, M. R. (2016). Gait analysis: Systems, technologies, and importance. Journal of Mechanics in Medicine and Biology, 16(07), 1630003.

Akhtaruzzaman, M., Shafie, A. A., & Khan, M. R. (2017). Quasi-inverse pendulum model of 12 DoF bipedal walking. International Journal of Automation and Computing, 14, 179-190.

Alekseev, D., Shagalova, P. and Sokolova, E. (2021), Development of a chatbot using machine learning algorithms to automate educational processes, GraphiCon 2021: 31st International Conference on Computer Graphics and Vision, September 27-30, Vol. 31, pp. 1104–1113.

Apu, M. A., Shakil, U. A., and Akhtaruzzaman, M., (2022). Development of an Intelligent Chatbot for MIST Website. CSE Technical Paper, The MIST Journal of Computer Science and Engineering, 1(1), 33-36.

Asha, N., Fiaz, A. S., Jayashree, J., Vijayashree, J., & Indumathi, J. (2022). Principal component analysis on face recognition using artificial firefirefly swarm optimization algorithm. Advances in Engineering Software, 174, 103296.

Augello, A., Gaglio, S., Infantino, I., Maniscalco, U., Pilato, G., & Vella, F. (2023). Roboception and adaptation in a cognitive robot. Robotics and Autonomous Systems, 164, 104400.

Belhumeur, P. N., Hespanha, J. P. and Kriegman, D. J. (1997), 'Eigenfaces vs. fisherfaces: Recognition using class specific linear projection', IEEE Transactions on pattern analysis and machine intelligence 19(7), 711–720.

Beymer, D. (1994), Face recognition under varying pose,", in 'Proceedings of 23rd Image Understanding Workshop', Vol. 2, pp. 837–842

Bishop, C. M. (1994), 'Neural networks and their applications', Review of scientific instruments 65(6), 1803–1832

Bogue, R. (2020), 'Humanoid robots from the past to the present', Industrial Robot: the international journal of robotics research and application 47(4), 465–472

Breazeal, C., Buchsbaum, D., Gray, J., Gatenby, D. and Blumberg, B. (2005), 'Learning from and about others: Towards using imitation to bootstrap the social understanding of others by robots', Artificial life 11(1-2), 31–62

Brooks, R. A. (1996), Prospects for human level intelligence for humanoid robots, in 'Proceedings of the First International Symposium on Humanoid Robots (HURO-96)', pp. 17–24

Chen, K., Yi, T., & Lv, Q. (2022). Fast and reliable probabilistic face embeddings based on constrained data uncertainty estimation. Image and Vision Computing, 121, 104429.

Cominelli, L., Hoegen, G. and De Rossi, D. (2021), 'Abel: integrating humanoid body, emotions, and time perception to investigate social interaction and human cognition', Applied Sciences 11(3), 1070

Dhamija, A., & Dubey, R. B. (2022). A novel active shape model-based DeepNeural network for age invariance face recognition. Journal of Visual Communication and Image Representation, 82, 103393.

Fong, T., Nourbakhsh, I. and Dautenhahn, K. (2003), 'A survey of socially interactive robots', Robotics and autonomous systems 42(3-4), 143–166.

Huang, P., Shen, Y., Yang, Z., Zhang, C., & Yang, G. (2022). Dual collaborative representation based discriminant projection for face recognition. Computers and Electrical Engineering, 102, 108281.

Iqbal, A., Shafie, A. A., Khan, M. R., Alias, M. F. and Radhi, J. (2011), 'Hri for interactive humanoid head amir-ii for visual tracking and servoing of human face', International Journal of Robotics and Automation (IJRA) 2(3), 220–23

Kanda, T., Ishiguro, H., Ono, T., Imai, M. and Nakatsu, R. (2002), Development and evaluation of an interactive humanoid robot" robovie", in 'Proceedings 2002 IEEE international conference on robotics and automation (Cat. No. 02CH37292)', Vol. 2, IEEE, pp. 1848–1855

Kotov, A., Arinkin, N., Zaidelman, L., & Zinina, A. (2020). Speech understanding system for emotional companion robots. Procedia Computer Science, 169, 63-69.

Li, L. (2023). Role of chatbots on gastroenterology: Let's chat about the future. Gastroenterology & Endoscopy, Elsevier. (Journal Pre-proof).

Li, Y., Liu, Y., & Wang, W. (2014). Planar hexagonal meshing for architecture. IEEE transactions on visualization and computer graphics, 21(1), 95-106.

Li, H. and Ngan, K. N. (2008), 'Saliency model-based face segmentation and tracking in head-and-shoulder video sequences', Journal of Visual Communication and Image Representation 19(5), 320–333.

Liu, C. C., Liao, M. G., Chang, C. H., & Lin, H. M. (2022). An analysis of children'interaction with an AI chatbot and its impact on their interest in reading. Computers & Education, 189, 104576.

Meena, D. and Sharan, R. (2016), An approach to face detection and recognition, in '2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE)', IEEE, pp. 1–6.

Milad, A., & Yurtkan, K. (2023). An integrated 3D model based face recognition method using synthesized facial expressions and poses for single image applications. Applied Nanoscience, 13(3), 1991-2001.

Nefian, A. V. and Hayes, M. H. (1998), Hidden markov models for face recognition, in 'Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)', Vol. 5, IEEE, pp. 2721–2724.

Rawal, N. and Stock-Homburg, R. M. (2022), 'Facial emotion expressions in human-robot interaction: A survey', International Journal of Social Robotics 14(7), 1583–1604.

Reggia, J. A., Katz, G. E. and Davis, G. P. (2018), 'Humanoid cognitive robots that learn by imitating: implications for consciousness studies', Frontiers in Robotics and AI 5, 1.

Rizomyliotis, I., Kastanakis, M. N., Giovanis, A., Konstantoulaki, K., & Kostopoulos, I. (2022). "How mAy I help you today?" The use of AI chatbots in small family businesses and the moderating role of customer affective commitment. Journal of Business Research, 153, 329-340.

Rojas-Quintero, J. and Rodríguez-Liñán, M. (2021), 'A literature review of sensor heads for humanoid robots', Robotics and Autonomous Systems 143, 103834.

Samaria, F. and Young, S. (1994), 'Hmm-based architecture for face identification', Image and vision computing 12(8), 537–543.

Sanaullah, M., Akhtaruzzaman, M., & Hossain, M. A. (2022). Land-robot technologies: The integration of cognitive systems in military and defense. NDC E-JOURNAL, 2(1), 123-156.

Sudha, S. S., & Suganya, S. S. (2023). On-road driver facial expression emotion recognition with parallel multi-verse optimizer (PMVO) and optical flow reconstruction for partial occlusion in internet of things (IoT). Measurement: Sensors, 26, 100711.

Turk, M. A. and Pentland, A. P. (1991), Face recognition using eigenfaces, in 'Proceedings. 1991 IEEE computer society conference on computer vision and pattern recognition', IEEE Computer Society, pp. 586–587

Wang, W., Liu, Y., Yan, D., Chan, B., Ling, R., & Sun, F. (2008). Hexagonal meshes with planar faces. Dept. of CS, HKU, Tech. Rep., 1-11

Wang, Y., Wang, L., Yang, T., Li, X., Zang, X., Zhu, M., Wang, K., Wu, D. and Zhu, H. (2014), 'Wearable and highly sensitive graphene strain sensors for human motion monitoring', Advanced Functional Materials 24(29), 4666–4670.