

Gesture Recognition System to Identify Animal Gestures

Amitha I C¹, Nanda S Nair², Nivedhya K C³, Thanusree Rajeevan C P⁴, Vyshna Pradeep⁵

¹ Asst. Professor, Computer Science and Engineering,

St Thomas College of Engineering and Technology, Mattannur, Kerala, India

²⁻⁵Computer Science and Engineering

St. Thomas College of Engineering and Technology, Mattannur, Kerala, India

Email: hodcs@stthomaskannur.ac.in, nandanair3@gmail.com, nivedhyakc@gmail.com, thanusreerajeev10@gmail.com, vyshnapradeep998@gmail.com

Abstract—Gesture recognition is a technique that lets us control any electronic device by identifying the movements or positions of different human body parts. Human gestures are a type of non-verbal interaction that can represent the most instinctive and unique approach for communicating with computers. When compared with all possible instances for gesture technology now available, there is no method for interpreting animal gestures. The objective of our project is to create an animal gesture recognition system that recognizes emotions of animals such as dogs and cats. The animal posture, such as the tail's position, hand movements and eye movements will be initially recovered and subsequently used for model training. A customized CNN method is used in the proposed system to classify the datasets. Currently there is no dataset available for validating the system. The user provides an image of the animal as input into the detector system. A dataset is created which contain images collected from the web is trained and processed and then comparison of input images is done. The corresponding classification of emotions such as anger, fear, happiness are conveyed by the gestures provided into the system displayed as the output. To get accurate results, features of images are used and after applying successful methodology for detection and recognition of the image, the system has obtained 95.6% accuracy using practical and user-friendly models and displayed as visuals.

Index Terms— Gestures, Convolutional Neural Network, Adam Optimizer, Emotions, ImageDataGenerator model, Animals

I. INTRODUCTION

The use of computer-based applications has increased in our daily life. Human-Computer Interaction will teach you how to accurately analyze and construct the interaction that exist between humans and computers. Applications for human-computer interaction are quickly across a range of industries, like artificial intelligence, image processing, IOT devices and so on. Artificial intelligence aims to imitate how the human brain works. A computational model that shows how the nerve cells work in human brain is called an artificial neural network. A class of artificial neural network technology where image detection and processing is done is called convolutional neural network. Face detection, emotion identification, and object detection are just a few of the many uses available with CNN. Gesture recognition facilitates nonverbal and wordless communication between humans and machines. There are various methods for recognizing gestures, including face and expression identification, hand

gesture recognition, and others.

The usage of gesture technologies has become widespread in the fields of robotics, artificial intelligence, machine learning and so on. It is a method that enables us to manage any electronic device by figuring out how various human body parts move or are positioned. Various sensor types are being identified and trained to control different applications. Human gestures are a particular kind of nonverbal method for interacting with computers when compared to all other prospective uses of gesture technology now available, there is no method for interpreting animal gestures. Animals communicate non verbally with one another and with people by using gestures, body language, and facial expression. Understanding various animal gestures tells us about the ways in which animals communicate their feelings to people

II. PROBLEM DEFINITION

We are familiar with many gestures shown by animals. These gestures explain various non-verbal ways of communication and the emotions expressed by them. But most of them are not recognizable to us. Recognizing various animal gestures teaches us about how animals express their emotions to humans. When compared with all the possible instances of gesture technology now available, there is no method for interpreting animal gestures. It is assumed that the existence and character of emotions in animals are similar to those in humans and have emerged from the same mechanisms. Our project's objective is to design a gesture recognition system that recognizes the emotions of animals. The Animal posture, such as the tail's position, hand movements, and eye movements, are initially recovered, and corresponding emotions are detected. The corresponding classification of emotions such as anger, fear, and happiness are conveyed by the gestures provided to the system displayed as the output. This study successfully achieves good recognition results.

These gestures describe a variety of non-verbal communication techniques as well as the emotions they convey. But most of the gestures are not recognizable to all. Recognizing various animal gestures teaches us about how animals express their emotions to humans. When compared with all the possible instances of gesture technology now available, there is no method for interpreting animal gestures. The nature and existence of emotions in animals are thought to be comparable to those in humans and to have developed from the same system.

Our project's primary goal is to develop a system for recognizing animal gestures that can recognize the emotions of animals like dogs and cats. We have chosen three fundamental emotions, such as anger, happiness, and fear for categorization. As there are currently no datasets available for verifying the system, we have produced a new dataset that has been processed and trained to compare the input images that the user has provided with the associated emotions displayed on the screen. To get good recognition results, we attempt to understand how these emotions are connected and associated in both people and animals. The entire project model is implemented using Python, HTML and CSS.

III. EXISTING SYSTEM

In the existing system, using a web camera, home appliances are controlled by identifying hand gestures. There are three methods; first is hand detection method ,using YCbCr skin segmentation method segmenting of unwanted background information is done and processing of noise is done to eliminate minor noises, which includes erosion, expansion processing, and smoothing morphological image processing. Second is hand tracking technique which uses KCF algorithm to track ROI and the third method is using Deep CNN Architecture for hand gesture recognition that combines traditional image processing with tracking. The disadvantage is the need of usage of depth detection networks.

IV. RELATED WORKS

Many researches has been conducted, but for finding out the emotions conveyed by the animals are not yet discovered. The primary goal is to identify the limits of prior studies and make recommendations for future research. Numerous algorithms using different methodologies have been created, but there hasn't been a literature review that compiles the methods that are currently in use.

A. Hybrid Approach for Facial Expression Recognition Using Convolutional Neural Networks and SVM

Here a combined model for facial expression with support vector machines (SVMs) and convolutional networks (CNNs) and their weighted summation is used for each frame the for the final facial expression classification. The system consists of three modules; the first module which composed of data collection, motion flow and facial landmark flow, the second module is facial motion flow classification which uses CNN and facial

geometry flow classification using SVM, and the third module is the classification of final outcome by using a mixture of CNN and SVM classifiers. Two datasets are used here; Cohn-Kanade (CK+) dataset and BU4D dataset. For ImageNet classification, a fundamental CNN architecture is utilized to enhance face expression recognition performance. There are two simultaneous process going on one is preprocessing training and the other is preprocessing testing and both these process have same working method; first an input image is normalized and landmark point is estimated and then motion flow is estimated and these values are fed into the two classifiers which provide a statistical result of seven facial expression classes and the resultant output is fused to find out weighted sum which is determined by the majority voting scheme. The major disadvantage is that it uses western database not eastern database and also limited congruency.

B. Gesture Recognition of Sign Language Using A Magnetic Positioning System

A magnetic positioning device has been developed to recognise static motions using the sign language alphabet. Techniques based on computer vision and sensor technology have both been applied in this case. The vision-based solution uses RGB or depth cameras to collect the images or video frames of the hand-performed signs. To recognise the signals and generate output in the form of text or speech, these must be further processed. The models are created with the aid of landmark data that was extracted from the dataset's images using MediaPipe, and these models are then validated with help from magnetic positioning system data. The technology uses a number of wearable transmitting nodes to measure the finger's 3D position and orientation. The measured location data is also processed using a machine learning technique. For the purpose of distinguishing 24 static letters from the ASL alphabets, the gesture is reconstructed using a method known as MagIK (magnetic and inverse kinematics). The letters J and Z are not taken into account when creating a machine-learning model because they are linked to dynamic gestures. The drawback is that it can only recognise static gestures.

C. Real-Time Hand Gesture Recognition System Using Fine- Tuned Convolutional Neural Network

A transfer learning methodology is used to resolve challenges such as lack of lack of labeled images in static hand gestures a deep cnn network must be trained. The Massey University collection includes 2515 color ASL static hand gesture pictures and the HUST-ASL dataset includes 5440 color photos of static hand motions and depth maps. CNN layers include convolution, pooling, fully connected, and outputs softmax. AlexNet and VGG-16 use Deep CNN architecture to categorize hand-gesture photos, while the dropout layer recognizes static hand motions and the output softmax layer predicts the necessary gesture class. The disadvantage is that a limited number of specialized shape-based feature extraction approaches are used.

D. Wearable-Sensors-Based Platform for Gesture Recognition of Autism Spectrum Disorder Children Using Machine Learning Algorithms

Here a classification technique for recognising autistic gestures using wearable sensors is being used. It transmits data to a data collection and classification server using Bluetooth from worn sensors. Ten autistic children repeat each gesture ten times to create a dataset of 24 gestures. Time and frequency domain features from sensor data are extracted, and these features are subsequently categorised using KNN, decision trees, neural networks, and random forest models. It is less feasible, which is the main disadvantage.

E. Automatic landmark point detection and tracking for human facial expressions

Here, a method for recognising and tracking facial landmarks in video recordings is proposed. Here, 26 landmark points on the facial region are found using scale-invariant feature-based detectors, making it easier to analyses human facial expressions. Several Differential Evolution-Markov Chain particle filters are used to track these areas across the video sequences. Testing has been conducted using three open-source datasets. The detection likelihood can be determined using a kernel correlation analysis approach by optimising the similarity criteria between the target points and the candidate points and incorporating it into the tracker's observation probability. For multiple DE-MC filters, the posterior distribution over the states of all target points is calculated using the M-component non parametric mixture model as opposed to a single DE-MC filter. The main flaw with this method is how ineffectively it detects and tracks objects.

F. Recognition of Finger spelling Sequences in Polish Sign Language Using Point Clouds Obtained from Depth Images

Here a technique for identifying static hand posture sequences as finger spelled Polish finger alphabet letters using point clouds is being proposed. Classification features such as VFH, eigenvalues-based descriptors, and modified versions of ESF and GRSD were used to classify the hand- shape representation. A hidden Markov

model classifier is a technique based here which is trained by transitions between postures corresponding to two consecutive letters. They used their dataset of testing and training data in the form of acronyms and letter transitions to achieve high identification rates for the acronym dictionary-dependent strategy. The disadvantage is that letter sequence classification without a dictionary.

G. A convolutional neural network with feature fusion for real-time hand posture recognition

Here, a convolutional neural network (CNN) and other conventional feature extractors are combined in a unique architecture to recognise hands accurately and in real time. On three different benchmark datasets, the suggested architecture is assessed and contrasted with convolutional neural networks. A lot of testing is done using binary, grayscale, and depth data along with two separate validation methods. It is demonstrated that the proposed feature fusion-based convolutional neural network (FFCNN) outperforms a variety of validation methods and image representations. When the aspect ratio of motions is retained, the recognition rate of FFCNN on binary images is equivalent to that of grayscale and depth. The main drawback is that there are no other ways of selecting and optimising hyperparameters.

H. Automatic Facial Expression Recognition Using Features of Salient Facial Patches

The framework for expression recognition proposed in this paper makes use of the visual characteristics of a few chosen facial patches. Depending on the location of facial landmarks, a few notable facial patches that are active during emotion elicitation are extracted. Different face patches are chosen as salient for various pairs of expression classes by further processing these active patches to get the salient patches, which contain discriminative information for the classification of each pair of expressions. Using these features, the one-against-one classification approach is used. Additionally, a suggested automated facial landmark detection method provides comparable results to previous landmark detection techniques. A solution for expression recognition in low-resolution photos is provided by the proposed method's ability to work well at various resolutions. The efficiency of the suggested system is demonstrated by tests on the CK+ and JAFFE facial expression databases. The primary drawbacks are that the analysis is limited to datasets without facial hair and that there is a probability that performance will improve with partially occluded photos.

I. A Kinect-Based Gesture Command Control Method for Human Action Imitations of Humanoid Robots

This study creates a humanoid robot that can learn human actions using a gesture command control system based on the Kinect. Kinect sensors are well-known for their excellent performance in gesture recognition. To fulfil the task of identifying the specified human active movements acquired by the Kinect sensor, this work employs three alternative mechanisms: dynamic time wrapping (DTW), hidden markov models (HMM), and principal component analysis (PCA)-based eigenspace techniques. Once the gesture command has been recognised, it is utilised to direct the actions of a humanoid robot, which will mimic the gesture command's content to imitate the active gesture of a human. The performance of three different detection algorithms for gesture commands is compared in a series of studies on gesture recognition and humanoid robot control, and the degree to which humanoid robot imitations of the gesture command-indicated human active gesture are similar is assessed. The main disadvantage is that it is less accurate.

After making a thorough survey, we contribute here a new detection model to find out the emotions of animals like dogs and cats shown through their gestures. This system has been divided into three components: a user and an admin interface, gesture detection and gesture recognition system. We employ a customised CNN model to operate this system, which can achieve appropriate identification accuracy while minimising computation load.

V. PROPOSED SYSTEM

The proposed system aims at classifying various emotions shown by dogs and cats using image processing techniques. A customized CNN method is used in the proposed system to classify the dataset. The model receives the user-provided image as input, which is then compared with the images that is already trained in the model. The output is a display of the corresponding emotions being expressed by the gestures.

A. Architecture

The architecture diagram of the system is given below in fig 1:

It consists of eight main components, which include input, pre-processing, region detection, feature extraction, feature database matching, classification, and output. The database block consists of various gesture images.

1) Input:- As soon as we login to the web page by providing a username and password, the user can input the image. The image is in the PNG format.

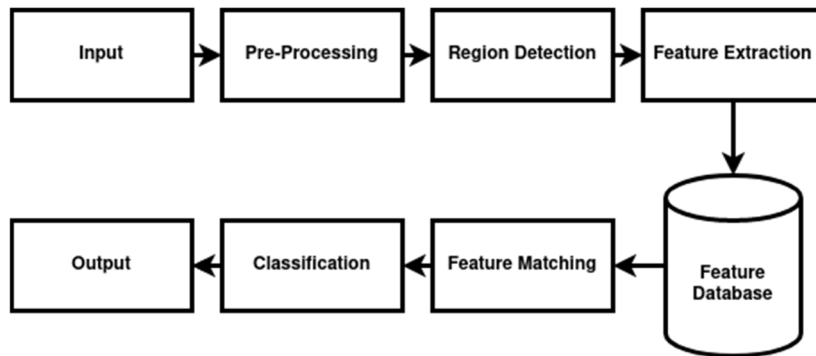


Figure 1. Architecture Diagram

2) Pre-processing:- The enhancement of image features and the subduing of unwanted distortions for further processing are done in preprocessing. The captured image is pre-processed to remove any noise that may affect the accuracy of the image processing step. This may involve applying various filters and techniques to enhance the image quality.

3) Region Detection:- Region detection is a phase in which the system detects all the important regions in the image from which features can be extracted using region detectors like edge detectors

4) Feature Extraction:- Feature extraction is the process of transforming raw data into numerical features that may be handled while preserving the original data set's information. It increases model accuracy by eliminating features from the input data.

5) Feature Database:- The feature database contains a variety of emotion gesture features that are trained and extracted and are thus used for comparison with the input image.

6) Feature Matching:- Feature matching refers to finding related features in two similar images.

7) Classification:- Image classification is the process of classifying and labelling a set of pixels or vectors inside an image based on specific characteristics. Structural details of the animals with similar characteristics are grouped together.

8) Output:- After conducting all the steps, the resultant images of emotions detected are shown, which include anger, happiness, and fear, and are used in the feature matching process to compare with the input image.

B. Gesture Detection

To identify the characteristics that the image describes, a convolution neural network is employed. The image is extracted, cropped, resized and converted in the grayscale. The animal images from a dataset is resized into 48x48 pixels, and trains the CNN model on the images. When reading an image with `cv2.imread()` and passing a flag value of 0, OpenCV uses the input picture to convert it to grayscale. The image file name is also passed as part of the operation. There is no pre-trained dataset available for the classification of emotions. Here a new dataset is created which consists of 600 images that includes emotions of dogs, cats for training and 400 images that includes all other images except cats and dogs for testing. The images in the dataset are classified into three categories of emotions consisting of anger, happy, fear of dogs as well as cats. These images in the data sets are converted into gray scale images using opencv library and are saved in a data list and the corresponding labels are stored in the label list. By performing a preprocessing stage, the image seeks to enhance some visual attributes that are crucial for subsequent processing or improve data to prevent unintended distortions. Pixel brightness are modified by a brightness transformation where the gray scale transformations without any regard to position in the image changes the brightness and also the colors are remained unchanged. A gray level interpolation, which assigns gray levels to the transformed image, changes the perspective of a given image for getting better about the required information. There are various interpolation techniques; in this case, we use bicubic interpolation, which enhances the model of the brightness function by locally approximating it by a bicubic polynomial surface; interpolation is done using sixteen nearby locations. By applying a kernel, which is a small array to each pixel and its neighbours inside an image, filtering is done to improve image qualities and to extract valuable information from the images, such as edges and corners. Basic filter techniques are used here for smoothing and sharpening of the images. Image resizing is also done in order to change the image's size without changing the amount of data in that image. Through region detection features are extracted in the subsequent stages which then detects all the important regions in that image. By performing feature extraction, raw data can be transformed into numerical features while keeping the original data's information. The model extracts

features from the images using convolutional layers, max pooling layers, and average pooling layers. These layers are used to identify patterns and features in the images that can be used to classify them.

C. Gesture Recognition

By applying feature matching, we may discover similar characteristics in two images. The brute-force method of feature matching compares the descriptors of every feature in the second image by calculating distances between each descriptor from each feature in the first image. The resulting pair then returns the nearest one. The process of categorising and labelling sets of pixels or vectors inside a picture is called classification, and it is carried out based on particular characteristics. The structural details of animals with similar characteristics are grouped together.

D. Convolutional Neural Network Model

A convolutional neural network (CNN) architecture using Keras is used to train the model. Batch processing technique is used here where instead of training the entire dataset at once, it is split into smaller subsets, or batches, and trained one batch at a time. To implement batch processing in the CNN structure provided, Fit method of the Sequential model and the batch size parameter is specified. The model architecture is defined using various layers such as Conv2D, MaxPooling2D, AveragePooling2D, Flatten, Dense and Dropout. The pooling layers reduce the spatial dimensions of the output feature maps, while the convolutional layers apply a collection of filters to extract features from the input images. On the basis of the retrieved features, the fully connected layers execute categorization. These layers define the structure of the neural network as well as its numerous parameters. The fit method is used to train the model on a specific dataset after the model architecture has been defined. The training data and labels for those data are input into the fit algorithm along with additional variables like batch size and epochs. The batch size option determines how many samples will be used in each training cycle. The number of iterations through the full training dataset is specified by the epochs option. In this case, the batch size is set to 32. The epochs parameter specifies the number of times to iterate over the entire training dataset. In this case, the model is trained for 35 epochs.

The CNN consists of two fully connected layers, three sets of convolutional and pooling layers, and a final output layer with softmax activation. A new sequential model object is made using the Sequential () method. The first layer in the model is a Conv2D() layer with 64 filters, a 5x5 filter size, and a ReLU activation function. The grayscale input images are 48x48 pixels in size, and the input form of the layer is set as (48, 48, 1). The MaxPooling2D () layer, which has a pool size of 5x5 and a stride of 2x2, is then applied to the layer's output to reduce its size by a factor of 2. The model then has two further convolutional layers with 64 filters each and 3x3 filter sizes in the following two layers, which are followed by an average pooling 2D () layer with a pool size of 3x3 and stride of 2x2. To extract more complex characteristics from the input images, this series of layers is created. The third set of convolutional and pooling layers is comparable to the second set, but the convolutional layers use 128 filters rather than 64. This group of layers is intended to draw out even more complex characteristics from the source images. After the output of the convolutional layers is flattened into a 1D vector using the Flatten () layer, it is then passed through two fully connected layers with 1024 units each and ReLU activation functions. Each completely connected layer is followed by a Dropout () layer to prevent overfitting. The output layer, which has num classes units and a softmax activation function to reflect the number of classes in the classification issue, is a Dense() layer. The layer's output values are guaranteed to accurately represent probabilities for each class via the softmax activation function.

E. Training

In order to increase the dataset and avoid overfitting, several transformations are applied to the input data during training using an ImageDataGenerator model that is created using the stated input data shape. For use with the training dataset, the ImageDataGenerator class in Keras offers a number of picture alteration techniques such random rotation, shifting, zooming, and flipping. These adjustments help the model become more resilient to changes in the input data by enabling it to learn to recognise objects from various angles, scales, and orientations. The fit generator() method, which generates batches of augmented data during training and aids in decreasing memory usage and boosting training performance, is used to train the model, and it is constructed using the flow() method. The Adam optimizer and a categorical cross-entropy loss function are then used to build the model. The Adam optimizer, a well-known stochastic optimisation algorithm used in deep learning, combines the advantages of both the Adaptive Gradient Algorithm and the Root Mean Square Propagation algorithms. It is used during training to update the model's weights based on the gradients of the loss function with respect to the weights. Because each input data point can belong to one or more classes, a multi-class

classification problem will develop, requiring the usage of the categorical cross-entropy loss function. It is saved as "model12.h5", the trained model. During training, the model adjusts its weights and biases to minimize the categorical cross-entropy loss function, which measures the difference between the predicted and true class probabilities. Table 1 below shows the training of the model sequential_1.

TABLE I. MODE: "SEQUENTIAL_L"

Layer (type)	Output Shape	Parameter
conv2d _1 (Conv2D)	(none, 44, 44, 64)	1664
max pooling2d _1 (MaxPooling2D)	(none, 20, 20, 64)	0
conv2d _2 (Conv2D)	(none, 18, 18, 64)	36928
conv2d _3 (Conv2D)	(none, 16, 16, 64)	36928
average pooling2d _1 (AveragePooling2D)	(none, 7, 7, 64)	0
conv2d _4 (Conv2D)	(none, 5, 5, 128)	73856
conv2d _5 (Conv2D)	(none, 3, 3, 128)	147584
average pooling2d _2 (AveragePooling2D)	(none, 1, 1, 128)	0
flatten _1 (Flatten)	(none, 128)	0
dense _1 (Dense)	(none, 1024)	132096
dropout _1 (Dropout)	(none, 1024)	0
dense _2 (Dense)	(none, 1024)	1049600
dropout _2 (Dropout)	(none, 1024)	0
dense _3 (Dense)	(none, 7)	7175

Total Parameters: 1485831

Trainable Parameters: 1485831

Non-Trainable Parameters: 0

V1. SOFTWARE USED

1) NumPy

It is a library for programming languages that adds support for large, multi-dimensional arrays and matrices, as well as a significant number of high level mathematical operations that may be performed on these arrays. It provides a broad library of high mathematical functions that work on these arrays and matrices, and also strong data structures that ensure efficient calculations with arrays and matrices.

2) Pandas

It is a Python-based software library with specialised data structures and operations for dealing with time series and mathematical tables that is intended for data manipulation and analysis. Merging is one of the data manipulation methods supported by Pandas.

3) Scikit-learn:-

It is a machine learning library for the Python language program which is employed in the implementation of statistical and machine learning models. Scikit-learn allows us to generate different machine learning models for regression, classification, clustering, and statistical tools for assessing these models.

4) Keras

A Python interface for artificial neural networks is provided by the open-source software framework Keras. Keras provides an interface for the TensorFlow library. It was made in Python and is used to quickly develop

neural networks. Moreover, different backend neural network computation is supported. Deep learning model distributed training also employs Keras.

5) *Tensor Flow*

An open-source, full-featured platform called TensorFlow is used to build applications for machine learning. It is a symbolic math toolkit that uses dataflow and differentiable programming to carry out a variety of tasks aimed at deep neural network inference and training. Developers can use a variety of tools, libraries, and community resources to create machine-learning applications.

6) *Open CV*

OpenCV is a cross-platform package which allows a user to create real-time computer vision applications. It focuses on areas like image processing, video capture, and analysis, which include tools for object and face detection. This library offers tools for processing and analyzing the content of images, which includes identifying objects in digital photos, tracking the movement of objects, converting images using machine learning techniques, and finding common elements in various images.

VI. RESULT

In this section, we have compared and analyze the results of the proposed convolution neural network model. Since there is no predefined dataset, a new dataset is created which is then trained and tested. There are 900 images in the dataset, where 600 images includes emotions of dogs, cats for training and 300 images that includes all other images except cats and dogs for testing. The system successfully classified the emotions of dogs and cats. The model accuracy obtained is 95.6%. The trained model is used for predicting the class labels in the dataset, and returns the predicted classes as a NumPy array. The model's predictions on the test dataset are computed using `model.predict classes()` and model's performance is evaluated on a test dataset by computing its confusion matrix using the `confusion matrix()` function from scikit-learn. The `load model()` function is used to load a pre-trained machine learning model from a file, and the `predict()` function is used to make predictions on the input dataset using the loaded model. We have used 80% of dataset for training stage and 20% of dataset for the testing stage. The neural network learns to make more accurate predictions and assigns higher probabilities to the correct class for each input data point by using categorical cross-entropy. The ImageDataGenerator model is used to prevent overfitting. Table 2 below shows the accuracy comparison of various models.

TABLE II. ACCURACY COMPARISON OF VARIOUS MODELS

Index	Title	Accuracy of model
1.	Hybrid Approach for Facial Expression Recognition Using Convolutional Neural Networks and SVM	94.69%
2.	Gesture Recognition of Sign Language Using A Magnetic Positioning System	95%
3.	Real-Time Hand Gesture Recognition System Using Fine-Tuned Convolutional Neural Network	92.1%
4.	Wearable-Sensors-Based Platform for Gesture Recognition of Autism Spectrum Disorder Children Using Machine Learning Algorithms	91.1%
5.	Automatic landmark point detection and tracking for human facial expressions	90.8%
6.	Recognition of Fingerspelling Sequences in Polish Sign Language Using Point Clouds Obtained from Depth Images	92.4%
7.	A convolutional neural network with feature fusion for real-time hand posture recognition	91.7%
8.	Automatic Facial Expression Recognition Using Features of Salient Facial Patches	93.2%
9.	A Kinect-Based Gesture Command Control Method for Human Action Imitations of Humanoid Robots	83.34%

VII. CONCLUSION

Gesture recognition refers to the ability of a computer to make out gestures and understand motion. The information or data gathered in this way may be used to execute further commands. In our proposed system, a

customised Convolutional Neural Network method for classification of image data set is used. The user inputs an image into the model, which compares it to the images already in the data set and the corresponding output is displayed with corresponding emotions being expressed by the gestures.

Our work also analyses the animal's entire image, identifies small changes in the animal's location, and gives a thorough understanding of the various emotions that the animals are expressing. A dataset is created where the images are stored as it is not available. It also understands and study about how the emotions are being conveyed by the selected animals through their gestures. To get accurate results, features are trained and after applying successful methodology for detection and recognition of the images, the system has obtained 95.6% accuracy using practical and user-friendly models and displayed as visuals.

In the future, this system can be improved by adding more emotions into the dataset for detecting animal gestures. Different animals can be used to find out their emotions. Further, for improving the performance of the model different techniques are applied.

REFERENCES

- [1] Rinalduzzi, M.; De Angelis, A.; Santoni, F.; Buchicchio, E.; Moschitta, A.; Carbone, P.; Bellitti, P.; Serpelloni, M. Gesture Recognition of Sign Language Alphabet Using a Magnetic Positioning System, *Appl. Sci.* 2021, 11, 5594.
- [2] Sahoo, J.P.; Prakash, A.J.; P lawiak, P.; Samantray, S Real-Time Hand Gesture Recognition Using Fine-Tuned Convolutional Neural Network Sensors 2022, 22, 706.
- [3] Chung, H.-Y., Chung, Y.-L., Tsai, W.-F. An Efficient Hand Gesture Recognition System Based on Deep CNN International Conference on Industrial Technology (ICIT).
- [4] Kim, J.-C.; Kim, M.-H.; Suh, H.-E.; Naseem, M.T.; Lee, C.-S. Hybrid Approach for Facial Expression Recognition Using Convolutional Neural Networks and SVM, *Appl. Sci.* 2022, 12, 5493.
- [5] Tie, Y., Guan, L. Automatic landmark point detection and tracking for human facial expressions, *EURASIP Journal on Image and Video Processing*, 2013(1).
- [6] Warchoń, D., Kapuściński, T., Wysocki, M. Recognition of Fingerspelling Sequences in Polish Sign Language Using Point Clouds Obtained from Depth Images., *Sensors*, 19(5), 1078
- [7] Chevtchenko, S.F.; Vale, R.F.; Macario, V.; Cordeiro, F.R. A convolutional neural network with feature fusion for real-time hand posture recognition., *Appl. Soft Comput.* 2018, 73, 748–766
- [8] Uzma Abid Siddiqui, Farman Ullah, Asif Iqbal, Ajmal Khan, Rehmat Ullah, Sheraz Paracha, Hassan Shahzad and Kyung-Sup Kwak Wearable-Sensors-Based Platform for Gesture Recognition of Autism Spectrum Disorder Children Using Machine Learning Algorithms *Sensors* 2021, 21(10), 3319;
- [9] Happy, S. L., Routray, A. Automatic facial expression recognition using features of salient facial patches *IEEE Transactions on Affective Computing*, 6(1), 1–12.
- [10] Ding, I.; Chang, C.; He, C. A kinect-based gesture command control method for human action imitations of humanoid robots. In *Proceedings of the 2014 International Conference on Fuzzy Theory and Its Applications (iFUZZY2014)*, Kaohsiung, Taiwan, 26–28 November 2014; pp.
- [11] Pugeault, N.; Bowden, R. Real-time ASL fingerspelling recognition In *Proceedings of the 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, Barcelona, Spain, 6–13 November 2011; pp. 1114–1119.
- [12] P. KaewTraKulPong and R. Bowden An improved adaptive background mixture model for realtime tracking with shadow detection In *Proc. The 2nd European Workshop on Advanced Video Based Surveillance Systems*, 2001.