Hand gesture classification using a novel CNN-crow search algorithm

Thippa Reddy Gadekallu¹ · Mamoun Alazab² · Rajesh Kaluri¹ · Praveen Kumar Reddy Maddikunta¹ · Sweta Bhattacharya¹ · Kuruva Lakshman¹ · Parimala M¹

Received: 6 November 2020 / Accepted: 1 March 2021 / Published online: 13 March 2021
© The Author(s) 2021

Abstract

Human–computer interaction (HCI) and related technologies focus on the implementation of interactive computational systems. The studies in HCI emphasize on system use, creation of new techniques that support user activities, access to information, and ensures seamless communication. The use of artificial intelligence and deep learning-based models has been extensive across various domains yielding state-of-the-art results. In the present study, a crow search-based convolution neural networks model has been implemented in gesture recognition pertaining to the HCI domain. The hand gesture dataset used in the study is a publicly available one, downloaded from Kaggle. In this work, a one-hot encoding technique is used to convert the categorical data values to binary form. This is followed by the implementation of a crow search algorithm (CSA) for selecting optimal hyper-parameters for training of dataset using the convolution neural networks. The irrelevant parameters are eliminated from consideration, which contributes towards enhancement of accuracy in classifying the hand gestures. The model generates 100 percent training and testing accuracy that justifies the superiority of the model against traditional state-of-the-art models.

Keywords Crow search algorithm · Hand gesture classification · Human–computer interaction · Convolution neural network · Meta-heuristic algorithms

Introduction

Since the advent of computers and digitization, Human–Computer Interaction (HCI) has become an intrinsic aspect

of Information studies. The terminology “human–computer interaction” was coined by [5] and popularly perceived as a study of man–machine interaction having an interdisciplinary potential of the application. HCI primarily incorporates the study of interface design, its usages emphasizing the interactions involving computers and its users. Since the computer is used in almost all facets of human life, the application of HCI is predominant across all verticals—computer science, psychology, social science, industrial engineering and many others [11]. Hand gesture is a type of non-verbal interaction technique which helps to provide the most intuitive and natural way of interacting with the computers. Hand gesture recognition plays a significant role in human computer interaction because the direct use of hands is the most natural and instinctive mode of communication among humans and also the present generation devices existing in the intelligent environment. Successful applications of hand gestures can be visualized in computerized game control systems, human–robot interactions and vision base recognition systems. Hand gestures also play a major role in interacting with devices such as smart homes, smart phones and various other gadgets wherein hands are used for communicating,

✉ Kuruva Lakshman
lakshman.kuruva@vit.ac.in

Thippa Reddy Gadekallu
thippareddy.g@vit.ac.in

Mamoun Alazab
alazab.m@ieee.org

Rajesh Kaluri
rajesh.kaluri@vit.ac.in

Praveen Kumar Reddy Maddikunta
praveenkumarreddy@vit.ac.in

Sweta Bhattacharya
sweta.b@vit.ac.in

Parimala M
parimala.m@vit.ac.in

¹ School of Information Technology and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India

² College of Engineering, IT and Environment, Charles Darwin University, Darwin, Australia

networking and interfacing with the environment [30]. There are four design approaches towards the implementation of HCI for developing efficient, user-friendly systems rendering instinctive user experiences. These four approaches are: “Anthropomorphic Approach,” “Cognitive Approach,” “Predictive Modeling Approach,” and “Empirical Approach”. These approaches can be used singly or as a combination in the design of an individual UI design [6]. The anthropomorphic approach in HCI helps in designing a user interface having qualities similar to humans. As an example, an interface could be designed such that it communicates with users similar to human-to-human interaction and would also display empathy in case of occurrences of exceptional events. In the case of the cognitive approach, the ability of human brains and sensory prediction mechanism is used to develop interfaces that fulfill user needs. Here, metaphors are used to depict abstract concepts and operations effectively to the users. As an example, a recycle bin icon is used to represent recycle bin in a PC. Although the name suggests “recycle bin,” but it does not recycle data in reality rather deletes files and the same concept is communicated effectively to the user using the icon. The Empirical approach is used for evaluating the usability of various conceptual designs. The testing is performed during the pre-production phase by balancing the concepts of designs and usability testings for each of the design concepts. The predictive modeling approach involves use of GOMS (Goal, Operators, Methods and Selection Rules) method to evaluate the components of a design based on the time it consumes to complete an interaction goal successfully [6]. GOMS is a human performance model that is used to enhance human–computer interaction by eliminating irrelevant and unnecessary interactive activities. The method uses a specialized human information processing model for increasing the efficiency of HCI describing four components of user’s cognitive structure. GOMS constitutes of a set of goals, operators, methods and selection rules in order to choose among alternative methods to achieve a desired goal. GOMS is extremely popular among computer system designers as it generates predictions on the usability of the system considering users perspective (Table 1).

Since the last four decades, almost all forms of human gestures have been studied and used either as a natural or intuitive method to interact with computational devices. To complement, all input–output technologies have been supportive of gesture oriented interactions. The use of gestures acts as a much more attractive yet effective alternative of complex interface devices for HCI. Gestures are as natural as computers completely integrated into our life. There exist various categories of gestures, as reviewed extensively in the literature. Deictic gestures focus on establishing the identity of an object’s spatial location within the limitation of the application domain, which includes a desktop computer, virtual reality application or a mobile device. Manipulative ges-

Table 1 Acronyms

HCI	Human–computer interaction
CSA	Crow search algorithm
GOMS	Goal, operators, methods and selection
CNN	Convolutional neural network
LSTM	Long short term memory
PCA	Principal component analysis
GRNN	General regression neural network
DCNN	Deep convolutional neural network
sEMG	Surface electromyography
EMG	Electromyography
ANN	Artificial neural network
MLP	Multi-layer perceptron
KP	Knowledge probability

tures are used to perform computational interaction with the sole purpose of controlling entities. The objective is to control the entities by establishing solid relationships between the actual movement of the hand gestures with the entities being manipulated. Semaphoric gestures are used for signaling with the help of flags, lamps, lights, and other indicators. It uses an organized dictionary of static and dynamic hand or arm gestures, which help to communicate with the machine. Gesticulations however, rely on computational analysis of actual hand actions relevant to the user’s speech content rather than pre-recorded mappings of gestures. Language gestures are different from other gesture styles performed based on a series of individual signs and conversation styles [21]. Deep learning and image processing are some of the most prominent technologies of today with an extremely bright prospective [13,19,32,43,48]. Gesture recognition is an application of the same technology. Studies have been performed to translate sign languages to alphabetic languages in real-time [8]. Gestures have the potential to convey semantic information and textual information pertinent to personality, emotion, or attitude. Studies have revealed that speech and gestures often share similar communication processes—also, gestures of an individual and their memory. Considering the same concept, the Convolutional neural network (CNN) has been used based on inter- and intra-parallel processing of the sequences in “hand-skeletal joints” for the classification of hand gestures. RGB image sequences and 3D skeletal data sequences both have been used for image processing purposes [10,50]. It is also evident that deep learning based models can be used effectively for image processing. As an example, 3D CNN models and Long Short-Term Memory (LSTM) recurrent networks have also been implemented using pre-computed image features and optical flows [28,44].

In [35], principal component analysis (PCA) and general regression neural network (GRNN) is used to develop a gesture recognition system. This system would be capable of

reducing signal dimensions, improve the accuracy and efficiency of real-time recognition. As part of the study, the key information relevant to human body motion are extracted to find specific action gestures and these gestures are used to extract features of the surface EMG. The PCA is applied to reduce the feature dimensions by eliminating irrelevant information for constructing GRNN neural network. This framework would help to identify the most accurate pattern of hand gesture leading to development of clinical medicine, healthcare prosthetics, HCI systems and various other systems.

The study in [1] utilizes the knowledge acquired from multiple modalities during the training of unimodal 3D convolutional neural networks (CNN) for hand gesture recognition. The framework involves devoting distinct networks for each modality and then integrating them to develop new networks having common semantics yet better in terms of accuracy and representations. The spatiotemporal semantic algorithm (SSA) helps to consolidate the feature contents from each of the distinct networks. The loss is handled using focal regularization parameter which ensures negative knowledge transfer is eliminated and performance of the system is enhanced yielding better test time recognition accuracy.

A two-antenna and Doppler radar-based approach is presented in [45] using deep convolutional neural network. The study has highlighted use of consumer radar embedded in circuits available in affordable prices, which are integrated with machine learning models [39,49] for smart sensing applications. The framework involves using a miniature sensor that captures Doppler signatures of 14 types of hand gestures which are further classified using a deep convolutional neural network. Two receiving antennas are of a continuous Doppler radars are used in the proposed model, capable of generating the in-phase and quadrature component of the beat signals. These signals are later mapped into three input channels of a DCCN which classifies the gestures with optimum accuracy with extremely low confusions between varieties of gestures.

Most implementations involving deep learning and image processing in gesture recognition use pre-trained models of CNN for feature extraction. However, the inclusion of an efficient feature engineering approach involving hyper-parameter tuning often remains ignored. Also, the choice of hyper-parameter tuning remains to be a major concern. The present study emphasizes these two aforementioned aspects, which acts as a motivation to identify the best feature engineering and hyper-parameter tuning approach that would yield better performance in gesture recognition in comparison to the existing studies. Thus, the motivation of the study includes:

1. Development of an efficient Convolutional Neural Networks model to achieve enhanced performance in gesture recognition

2. Use of crow search optimization method to select most accurate combination of hyperparameters that would contribute to fulfilling the desired accuracy in results.

The proposed method focuses on fulfilling the aforementioned objectives. The first step involves accessing the first-hand gesture image dataset from the publicly available Kaggle dataset. Next, one hot encoding is performed to convert categorical data to binary values, for making the dataset fit for processing by the CNN. Crow search algorithm is then implemented for hyper-parameter optimization, and the resultant hyperparameters are fed into CNN to achieve the desired output. The model is finally evaluated against the state of the art models and the results clearly justify the superiority of the model. Hence, the unique contribution of the paper definitely highlights the use of crow search algorithms for hyper-parameter tuning of the parameters. The algorithm is one of the most popular algorithms used to resolve optimization problems considering the minimum control parameter, which is also the reason behind its success in delivering the best accuracy in minimum time consumption.

The unique contributions of the proposed framework are:

1. The application of crow search metaheuristic algorithm (CSA) to choose the optimal hyper-parameters for training the data in the CNN.
2. An accuracy of 100% is achieved on the hand gesture dataset which is superior to the existing state-of-the-art works.

The organization of the paper is as follows. Section 2 presents an extensive survey of the existing work done in this domain of research. Section 3 provides background knowledge of the subject area and also describes the proposed architecture. Section 4 highlights the results of experiments and incorporates the conclusions drawn.

Literature survey

Kamal et al. [29] proposed a pattern recognition method for static recognition, which is able to handle the low variability among the different gestures. Authors have used shape geodesics and robust registration for calculating the accelerated time. The proposed system is evaluated by considering three distances of the shape geodesics, and the experiment results showed that the proposed model is efficient than the other related methods.

Wei et al. [51] proposed a multi-view deep learning model by relating classical surface electromyography (sEMG) feature sets with a CNN-based deep learning model to recognize the gestures. The multi-view model mainly emphasized on the parallel functioning of CNN multi-streams and training

of the network with deep feature sets of sEMG gestures. Experiments were conducted with 11 different databases of sEMG, and results shown that the multi-view model performs exceptionally well on the dissimilar data streams of sEMG.

Tan et al. [47] proposed a static gesture recognition model using electromagnetic fields. This model primarily focuses on vision-based recognition and provides training with CNN by an end-to-end recognizer. The proposed model was tested with the various datasets of static hand gesture images and achieved 99% recognition rate for full aperture, and for one-eight aperture, the accuracy is 95.32%. Results outperformed even for the limited aperture and also had improved scalability on the gesture images.

Hu et al. [14] proposed a hand gesture recognition system to control the unmanned aerial vehicles (UAV). The entire model has been trained and tested with the various layers of deep learning neural networks like 2-layer and 5-layer fully connected neural network and a CNN of 8 layers. The experimental results proved that the efficiency is better than the existing systems and achieved an average accuracy of 96.7% for 2 layers and 98% for 5 layers. Finally, CNN with 8 layers attained 89.6% and 96.9% on scaled and non-scaled datasets.

Okan et al. [22] proposed a model that works for the video hand gesture recognition. CNN is used to classify and detect the number of gestures and also in evaluating single-time activations. Two datasets NVIDIA and EgoGesture were used in calculating the efficiency of the gestures and achieved an accuracy of 94.03 %. The model was very well extended for the sliding window approach, and the results are outperformed compared to the existing video recognition systems.

Sruthy et al. [45] proposed a CNN-based hand gesture recognition framework for capturing various hand gestures. The deep convolutional neural network [25,26,50] used in this work to classify the gestures and train the two spectrograms of the Doppler radar capable. The proposed model got trained by CNN, and testing was done in two phases in producing the quadrature components. The experimental results proved that the proposed architecture has a good accuracy of 95% compared to the other models.

Pinto et al. [33] proposed a gesture recognition based model using convolutional neural networks. This method mainly focuses on the preprocessing steps like polygon filter and segmentation process of the various gestures. Using convolutional neural networks, the training and testing part has been carried out by 60% and 40%. The results are analyzed both in the testing and training processes, and the calculated metrics show that the proposed model is robust than the existing methodologies.

Li et al. [24] proposed CNN-based hand gesture recognition framework where the number of gestures is characterized

by the neural network and error backpropagation algorithm. In this model, the recognition of gestures and extracting its features were labeled by unsupervised learning approaches. Further, support vector machine was considered to examine the best possible gestures from the optimized dataset. It has proved that the proposed system shows a high accuracy by means of classification of gestures in static and dynamic representation.

Ahmed et al. [2] proposed a novel method of recognition of gestures by finger counting using convolutional neural networks. It provides an immersive experience to the gesture handling people, and researchers used it for an alternative approach in accessing the optimal location of a gesture recognizer. Proposed model impulses the finger counting and labels to the sensors and motions of a human body. This model gives better accuracy over the other frameworks and performs a stable recognition for real-world applications.

Jiang et al. [18] proposed a vision based recognition method using convolutional neural networks. It aims to perform the best possible hand gestures of a human body by means of keletonization algorithm and CNN. Here, the gesture recognition process was carried out by the spatial coordinate system and sparse representation. The model has been trained and tested by the American Sign Language database and the results showed that the proposed model is having a high recognition rate of 96.01% to the existing frameworks.

Jinxian et al. [35] proposed a system to identify hand gestures using EMG signals and PCA and Generalized regression neural network (GRNN). The model is processed with nine static gestures and extracted the important human emotions. It is further improvised for the real-time recognition of human emotions and reduced the signal dimension. Finally, the proposed model showed overall recognition rate as 95% after dimensionality reduction and training with neural network and gave the better average recognition compared to the existing approaches.

Chen et al. [7] proposed a deep neural network model for recognizing the hand gestures using CNN through surface electromyography signals. As the proposed model progress the accuracy in classification and also diminishes the various parameters compared to the existing hand gesture recognition methods. Classification accuracy process was done by the classical machine learning methods and executed on the Dataset Myo. Further, the model provides better results with sEMG signals and also provides the classification of sEMG signals along with the CNN architecture.

Research works represented in the literature review are summarized with the methods used, key findings and limitations are addressed in the Table 2.

Table 2 Summary of the research works used in Literature Review

Paper cited	Methods used	Key findings	Limitations
Kamal et al. [29]	Pattern recognition for static recognition	Proposed system is evaluated by considering three distances of the shape geodesics	Applicable only to static gestures and not able to handle dynamic gestures.
Wei et al. [51]	CNN-based deep learning model	The multi-view model mainly emphasized on the parallel functioning of CNN multi-streams and training of the network with deep feature sets of sEMG gestures.	Experiments were conducted with 11 different databases
Tan et al. [47]	Vision-based recognition with Neural Network	Proposed model was tested with the various datasets of static hand gesture images and achieved 99% recognition rate for full aperture	Model only focuses on the static gestures.
Hu et al. [14]	CNN with unmanned aerial vehicles (UAV)	The entire model has been trained and tested with the various layers of deep learning neural networks like 2-layer and 5-layer fully connected neural network and a CNN of 8-layer.	This model is mainly applicable for non-scaled datasets.
Okan et al. [22]	Backpropagation Neural Network and PCA	It works for the video hand gesture recognition. Two datasets NVIDIA and EgoGesture were used in calculating the efficiency of the gestures and achieved an accuracy of 94.03 %.	Datasets with less instances are considered.
Sruthy et al. [45]	Deep convolutional neural network	Deep Convolutional network is used in this work to classify the gestures and train the two spectrograms of the Doppler radar capable.	Features in the dataset are very less and also not tested with a huge data items.
Pinto et al. [33]	Convolutional neural networks	This method mainly focuses on the preprocessing steps like polygon filter and segmentation process of the various gestures.	Extraction process was considered with a very less number of features.
Li et al. [24]	CNN & NN & Error backpropagation algorithm	Recognition of gestures and extracting its features were labeled by unsupervised learning approaches.	Classification done only for limited images.
Ahmed et al. [2]	CNN & Neural Network (NN)	Proposed model impules the finger counting and labels to the sensors and motions of a human body.	Less accuracy in the early stages and average recognition rate is low.
Jiang et al. [18]	CNN & Keletonization algorithm	Gesture recognition process was carried out by the spatial coordinate system and sparse representation.	Very few samples were considered for extraction process.
Jinxian et al. [35]	Principal Component Analysis (PCA) & GRNN	Classification assessment process done for real-time human emotions and achieved overall recognition rate as 95%.	Used only nine static gestures and not used any dynamic gestures.
Chen et al. [7]	Deep Neural Network	Extracted features and achieved better results with sEMG signals.	Classification accuracy process was only done by the classical machine learning methods.

Background and proposed architecture

In this section, Convolution Neural Networks and Crow Search optimization algorithms are discussed, followed by the architecture of the proposed model.

Convolution neural network (CNN)

Here, we discuss the general structure of CNN along with the different types of optimization functions, epochs, batch size, convolutional layers, pool layers, dense layers, loss functions, activation functions.

The Convolutional Neural Network (CNN) is the most popular network for image analysis, data analysis, and classification problems [25,31]. Generally, CNN is an artificial neural network that specializes in being able to pick or detect patterns and make sense of them. Pattern detection makes CNN so useful in image analysis. CNN is a form of an ANN that makes it different from a standard Multi-Layer Perceptron (MLP) [54]. CNN has a hidden layer called convolution layers, and more precisely, these layers are able to detect patterns by specifying the number of filters in each layer [17,25,31,37,54]. CNN has other layers of non-convolution, but the basis of CNN is the layers of convolution [17]. The purpose of the convolution layers is to receive the input and then output the transform input to the next layer, and this transformation is a convolution operation which is given in Fig. 1.

Zero padding

When a filter transforms input data, it tends to output as a matrix. The dimensions of the image are changed during this process. The main purpose of zero padding is to add zeros to the matrix to adjust the image as required. Zero padding is primarily used to compute highly interpolated spectra by considering the Discrete Fourier Transformation (DFT) of the zero padded signal. This type of interpolation is applicable when the original signal is time limited. Zero padding is predominantly used for analyzing data from the non-periodic signals existing in the blocks. Here, each block or signal is considered as a finite-duration signal which is zero padded on either side with any number of zeros. This zero padding has the potential to yield more denser interpolation of the frequency samples around the unit circle.

Dense layers

The neurons in the layers are compactly connected to all previous layer neurons [42]. The key benefit of the dense layer is that neurons linked in layers have different combinations of features from previous layers.

Polling

Pooling is another key element of CNN that is imposed between the convolution layers to reduce the spatial size of the data, boost the computation of the network and minimize over-fitting. There are two types of pooling available, namely Max Pooling and Average Pooling. Max pooling picks the maximum value in the area of the feature map, whereas the average pooling selects the average value of the Map feature.

Activation function

Activation functions are computational equations that quantify the performance of a NN. These activation functions perform complicated calculations on hidden layers and transfer to the output layer. Activation functions are primarily intended to create non-linear features in the NN [38]. The function is associated with every neuron in the network and evaluates whether to enable or disable the neurons. Activation function normalizes the output value of each neuron within the range $[1,0]$ or $[-1,1]$. There are seven types of DNN activation functions, namely Sigmoid, TanH, ReLU, Leaky ReLU, Parametric ReLU, Softmax, Swish.

Optimization functions

Optimizer algorithms are used to fine-tune the NN's properties, which include updating weight and learning rates, to minimize losses and converge in a minimum amount of time that leads to better performance [12,27]. There are different types of optimizers in DNN, namely Gradient Descent, Stochastic Gradient Descent, Mini-Batch Gradient Descent, Nesterov Accelerated Gradient, Adagrad, AdaDelta and Adam.

Loss functions

During the training of the NN, the loss is defined as finding an error in the NN, and the function used to predict the error is called Loss function. There are different types of loss functions available, but identifying the appropriate loss function to predict loss is a challenging task. Some of the loss features available are Mean Squared Error, Binary Cross entropy, Categorical Cross entropy, Sparse Categorical Cross entropy.

Epoch

The total number of epochs determines how many cycles the algorithm will perform on a training dataset. One epoch means that all available samples in the training dataset will be given the opportunity to update their weight. The total number of epochs depends on the rate of error and the weight updating.

Fig. 1 Fully connected CNN

Batch size

The number of samples that will be passed through the network at one iteration is defined as batch size. The batch size can be represented in three ways, namely batch mode, mini-batch mode, stochastic mode. In batch mode, the batch size is equivalent to the size of the whole dataset, where the number of iterations and epoch values are the same. In mini-batch mode, the batch size is less than the dataset. Finally, in stochastic mode, the batch is equivalent to one.

Crow search algorithm (CSA)

CSA is one of the recent meta-heuristic algorithms [16,46]. Crows are considered the smartest birds which have the biggest brain compared to the size of the body. There are plenty of proofs to prove the crows are very clever. They displayed self-awareness in mirror tests and tool-making skills, and also they can recall faces easily. In addition, they can use tools, communicate well, and remember their food until some months later [3,34,40].

Crows were thought to look at other birds, identify the place where other birds hide, and snatch their food as soon as the owner leaves them. It will take extra measures to prevent becoming a potential victims if its committing to robbery. In fact, it uses its own experience as a thief to predict pilferer actions and can decide the best way to prevent proliferation of its caches [9]. Flock form remembers the hiding places, following the other to do the robbery and protecting their things from a steal by chance are the properties of CSA.

It is assumed that there are several crows in a d -dimensional environment. The number of crows is N (size of flock) and the location of the crow j on the $rept$ (repetition) search space is vector-specified $Y^{j, rept}$ ($j =$

$1; 2; \dots; N; rept = 1; 2; \dots; rept_{max}$), where $Y^{j, rept} = [Y_1^{j, rept}, Y_2^{j, rept}, \dots, Y_d^{j, rept}]$ and maximum number of repetitions is $rept_{max}$. $n^{j, rept}$ shows the position of the crow j hiding place in the repetition $rept$. the crow j has achieved the best position. Crows are running about and finding better sources of food (hiding places).

Suppose that crow j would like to visit her hiding place during the iteration $n^{j, rept}$. Crow j wants to follow crow k to get the hiding place of crow k in this iteration. Two states can occur in this case:

State 1 The crow j will identify the crow k hiding place without knowledge of crow k . The new position of the crow j is achieved as follows in this case:

$$Y^{j, rept+1} = Y^{j, rept} + s_j \times flen^{j, rept} \times (n^{k, rept} - Y^{j, rept}), \quad (1)$$

where random number with uniform distribution between 0 and 1 is s_j and $flen^{j, rept}$ represents the flight length of crow j at repetition $rept$.

State 2 The crow k can fool crow j by going to a different search space position to protect its cache from being pilfered if it knows that crow j following it.

Totally, the following states 1 and 2 can be expressed:

$$Y^{j, rept+1} = \begin{cases} Y^{j, rept} + s_j \times flen^{j, rept} \times (n^{k, rept} - Y^{j, rept}) & s_k \geq KP^{k, rept} \\ \text{a random position} & \text{otherwise} \end{cases} \quad (2)$$

The diversification and intensification should be well balanced by meta-heuristic algorithms [52]. Diversification and intensification are two major components of any meta-

heuristic algorithms. Diversification refers to the capability of the algorithm to generate diverse solutions by exploring the search space in a global scale. On the contrary, intensification refers to focusing the search activity within the local space while being aware that the solution would be found in the local search space itself. The balance of both ensures that the best solution and global optima is achieved ensuring improvement in the convergence rate. The CSA mainly monitors intensification and diversification by the knowledge probability (KP) parameter. As the probability value of knowledge decreases, CSA tends to search for a local area where the best solution in this region exists. It increases the intensification by using low KP values. In addition, the chance to search near current successful solutions decreases by increasing the KP, and CSA prefers to explore the global search field (randomization). It improves diversification in the use of big KP values.

The step-by-step process for implementing CSA is as follows:

1. The adjustable parameters of CSA (flock size (N), maximum number of repetitions(reptmax), length of flight (flen) and knowledge probability (KP)) are valued.
2. In a-dimensional search field, N crows are randomly placed as members of the flock. A feasible solution is indicated by each crow, and a is the number of decision variables.

$$\text{Crows} = \begin{bmatrix} Y_1^1 & Y_2^1 & \dots & Y_a^1 \\ Y_1^2 & Y_2^2 & \dots & Y_a^2 \\ \vdots & \vdots & \vdots & \vdots \\ Y_1^N & Y_2^N & \dots & Y_a^N \end{bmatrix}. \quad (3)$$

Here, the memory of each crow is initialized. Because the crows do not have experience at the initial iteration, their food at the first positions is believed to be disappeared.

$$\text{Memory} = \begin{bmatrix} mr_1^1 & mr_2^1 & \dots & mr_a^1 \\ mr_1^2 & mr_2^2 & \dots & mr_a^2 \\ \vdots & \vdots & \vdots & \vdots \\ mr_1^N & mr_2^N & \dots & mr_a^N \end{bmatrix}. \quad (4)$$

3. The standard of its position shall be determined for each crow by adding it in the objective function of the decision variable value.
4. In the search space, crows establish the new location as follows: Assume that crow j wants to establish a new location. For this purpose, this crow selects a crow randomly (e.g., crow k) to see how the food is caught by this crow (m^j). In eq. (2), the new location of the crow i is achieved. This method applied for all crows in the field.

5. Need to check the stability of all crow's new locations. The crow updates its new location if the new location of the crow is stable. Otherwise, the crow would stay in the current position and do not shift to the new position created.
6. The fitness value for every crow's new location is determined.
7. The memory of the each crow is updated as follows:

$$mr^{i, \text{rept}+1} = \begin{cases} Y^{j, \text{rept}+1} & \text{if } \text{flen}(Y^{j, \text{rept}+1}) \geq \text{flen}(mr^{j, \text{rept}}) \\ mr^{j, \text{rept}} & \text{0. W.} \end{cases}, \quad (5)$$

where flen represents the value of objective function. The crow updates the new location in its memory if the fitness value of the new location is better than the fitness value of the remembered position.

8. Steps 4–7 will repeat till getting the $\text{rept}_{\max}$. The best location of the memory with regard to the objective function value as the solution to the problem of optimization [23] shall be indicated when the termination criterion is met.

Algorithm 1: Crow search algorithm

```

Initialize the location of all  $N$  crows in the search field;
Calculate the location of the crows;
Initialize the memory of each crow;
while  $\text{rept} \leq \text{rept}_{\max}$  do
  for  $j=1:N$  (all  $N$  crows of the flock) do
    Randomly choose one of the crows to follow (for example  $k$ )
    Define an Knowledge probability if  $s_k \geq K P^{k, \text{rept}}$ 
    then
       $Y^{j, \text{rept}+1} = Y^{j, \text{rept}} + s_j \times \text{flen}^{j, \text{rept}} \times (m^{k, \text{rept}} - Y^{j, \text{rept}})$ 
    else
       $Y^{j, \text{rept}+1} =$  a random position of search space
    Check the feasibility of new locations;
    Calculate the new location of the crows;
    Crows memory update;
  end for
end while

```

The crow search algorithm (CSA) is an extremely efficient algorithm for finding optimal solution in the search space. The advantages of CSA include its simple implementation, use of few parameters and flexibility. The study in [15] has performed a comparative analysis of CSA with various other meta-heuristic algorithms namely Grey Wolf Optimization, Particle Swarm Optimization, Sine Cosine Algorithm, Bat Algorithm, etc. The Friedman test was conducted in [15] and the results of the evaluation have justified the significance of CSA over the other meta-heuristic algorithms. But there does

exist some scalability issues with CSA in cases of handling multi-modal data yielding in low convergence rate. Hence CSA is fine tuned, modified or hybridized into three types of classes namely variants, hybrid and multi-objective which has further improved its efficiency.

Proposed architecture

Several hyper-parameters such as number of convolution layers, number of dense layers, pooling layers, optimization function, activation function, number of epochs, the batch size for iteration, loss function have to be passed to the CNN. Choosing the right combination of these hyper-parameters (Hyper-parameter tuning) is vital to achieve better performance. Hyper-parameter tuning is an NP-Hard problem, which makes it very difficult to choose the right value for each of the parameters. The typical metaheuristic algorithms use permutations to solve NP-hard problems. But CSA does not directly generate permutations. It uses continuous number encoding technique for computing a swarm-based metaheuristic representation. Even though several hyper-parameter optimization approaches like grid search [4], random search and Bayesian optimization approaches exist for hyper-parameter tuning, their performance dips when the number of hyper-parameters are huge. In grid search, an extensive search is conducted for the selection of a model. The data scientists prepare a grid of hyper-parameter values and for each combination, the model is trained and scored based on the testing data. All possible combination of hyper-parameter values are tried and hence the algorithm becomes extremely inefficient. In case of Random search, a grid of hyper-parameter values are set up and random combinations are selected to train and score the model. Hence, the number of parameter combinations to be attempted can be explicitly controlled which enhances its efficiency. Nature-inspired algorithms can play a very effective role in hyper-parameter tuning as they can significantly reduce the search space and find the optimal solutions by global optimizers [36,41,53]. The use of nature inspired algorithms have been extremely predominant in various applications but its has its associated challenges from theoretical views. Although the basic functioning of the algorithm is well understood but the reason and associated condition of its functioning often lacks clarity. These algorithms also have their own algorithm dependent parameters wherein the value of the parameters affect its performance when trying to achieve optimum performance. Due to its fast convergence rate, high efficiency and few control parameters, the crow search algorithm is chosen in this work for tuning the hyper-parameters.

The proposed model is depicted in Fig. 2. The steps in the proposed model are summarized as follows:

- The hand gesture image dataset is loaded from kaggle.

Fig. 2 Proposed model

- **Apply one-hot encoding**—Machine learning algorithms can not process categorical data. One-hot encoding is used in this work to convert the labels from categorical into binary values which can be processed by the CNN. In case of categorical variables, ordinal relationships do not exist and thus, integer encoding appears to be insufficient. Using integer encoding and making the model assume natural ordering between the various categories results in inferior performances. One hot encoding technique is applicable in such cases to the integer representation. Here, the integer encoded variable is eliminated and a new binary variable is added for each of the unique integer value.
- Identify the new location of the crow using Eq. 1
- Using Eq. 3, locations and memory initialized
- The memory of each crow is updated using Eq. 5, where f_{len} represents the fitness function value, which is used for hyperparameter tuning in CNN.
- Based on the obtained hyperparameters, the dataset is trained with the help of CNN.
- The results obtained from the proposed crow search-based approach are then compared with other CNN models based state-of-the-art nature-inspired algorithms such as Whale Optimization Algorithm (WOA), Gray Wolf Optimization (GWO), Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Gravitational Search Algorithm (GSA), Ant Bee Colony (ABC) algorithm and Cuckoo Search Algorithm (CSA).

Results and discussion

The experimentation was performed on a publicly available dataset collected from Kaggle. For the experimentation purpose, we have used “Google Colab”, the GPU-based cloud framework offered by Google Inc. This framework had 50 GB Hard Disk and 25 GB RAM. The Google Colab is an online browser-based platform that enables data scientists to train the models on machines without any expenses. Since it uses the computational power of the google servers instead of the users machine, the performance is enhanced saving time for computation as well. The programming language used is Python 3.7. The following subsections discuss the dataset description and the performance evaluation of the proposed model.

Fig. 3 Sample images from the dataset **Table 3** The hyper-parameters of the CNN chosen by the CS algorithm

Hyper-parameter	Description
Optimizer	Adam optimizer
Numer of epochs	5
Batch size	64
Number of conv. layers	3
Number of pool layers	3
Number of dense layers	2 (One intermediate and one output layer)
Loss function	Categorical and crossentropy
Activation function at intermediate layer	Relu
Activation function at output layer	Softmax

Fig. 4 Accuracy of the proposed model

Fig. 5 Validation loss of the proposed model

Dataset description

The dataset used for this experimentation, “Hand gesture recognition database,” was collected from the public repository, Kaggle [20]. has 10 different folders for hand gesture images for 10 digits (0–9). Each folder has 2000 collection of images for different hand gestures for the corresponding digits. Few sample images from the dataset are depicted in Fig. 3.

Experimental setup

In this work, 80% of the images were used for training and 20% of the images were used for validation. CS optimization was used to choose the hyper-parameters for CNN. The hyper-parameters of the CNN chosen by the CS algorithm are as shown in Table 3.

Performance evaluation of the proposed model

This subsection discusses the performance evaluation of the proposed model. The metrics used to evaluate the proposed model are accuracy and loss. Figure 4 depicts the performance of the proposed model based on the accuracy metric. From this figure, it can be observed that at the end of 3rd epoch, both training and testing accuracy are 100%. Similarly Fig. 7 depicts the loss rate of the proposed model based on number of epochs. From the figure, it is evident that by the end of the 3rd epoch, both training and testing loss becomes 0%.

The accuracy of the proposed crow search-CNN model is then compared to CNN models integrated with WOA, GWO, PSO, GA, GSA, ABC and CSA algorithms. Figure 6 depicts the comparative results. From the figure, it is evident that the proposed crow search-based CNN outperforms the considered models with training and testing accuracy of 100%.

Fig. 6 Performance evaluation based on training and testing accuracy

The loss rate of the proposed algorithm is then compared with other nature-inspired based CNN models. The results are depicted in Fig. 7. From the figure, it is evident that the proposed model achieved a loss rate of 0%, thus outperforming the other models considered.

Figure 8 depicts the performance evaluation of the proposed model with other models based on training time. From the figure, it can be observed that the proposed crow search-based approach trains CNN in 16 min, which is very less compared to the other considered models.

Discussion

The crow search algorithm is one of the popular nature-inspired algorithms used for many optimization problems. The crow search algorithm has advantages such as fast convergence rate and considers very few control parameters. These features of the crow search algorithm make it an apt

Fig. 7 Performance evaluation based on training and testing loss

Fig. 8 Performance evaluation based on training-time

choice for tuning the hyper-parameters of the CNN. From the results, it can be observed that the proposed crow search model has performed better than the state-of-the-art nature-inspired algorithms in tuning the parameters of the CNN. The results achieved can be summarized as follows:

- The proposed CSA-CNN model outperformed the other state-of-the-art nature-inspired algorithms in terms of training and testing accuracy and loss.
- The training time of the proposed model is less than the other models considered.

Conclusion

The present study introduces a new framework for hand gesture recognition based on convolutional neural networks. Deep learning and CNN based models are quite popular approaches in gesture recognition. Choosing the right hyper-

parameters for CNN plays a vital role in achieving the expected classification results. The present study focuses on choosing the optimal hyper-parameters of the CNN to classify publicly available hand gesture dataset from Kaggle. First, one-hot encoding technique is applied on the dataset to transform categorical values to binary format. Then, crow search meta-heuristic algorithm is used for choosing the optimal hyper-parameters for the CNN. Then, the CNN is trained on the resultant dataset using the hyper-parameters chosen by CSA algorithm. The classification results generated are evaluated against the state-of-the-art models. The performance evaluation shows 100 percent training and testing accuracy results utilizing only 16 min of training time which outperforms the existing approaches. As highlighted in the paper, crow search algorithm is a meta-heuristic model which is derived from the behavior of crows, especially their act of searching food. Although CSA has its benefits when implemented in CNN frameworks, but the search strategy involved in this method has its associated issues when subjected to high multi-modal formulations. Considering this challenge, the future direction of research lies in improving the convergence in case of high multi-model formulations. This improvised version of CSA could be implemented of real-time larger dataset wherein the performance could be practically analyzed and validated.

Declarations

Conflict of interest The authors declare that they have no conflict of interests to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Abavisani M, Joze HRV, Patel VM (2019) Improving the performance of unimodal dynamic hand-gesture recognition with multimodal training. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1165–1174
2. Ahmed S, Khan F, Ghaffar A, Hussain F, Cho SH (2019) Finger-counting-based gesture recognition within cars using impulse radar with convolutional neural network. *Sensors* 19(6):1429

3. Askarzadeh A (2016) A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. *Comput Struct* 169:1–12
4. Bhattacharya S, Kaluri R, Singh S, Alazab M, Tariq U et al (2020) A novel PCA-firefly based XGBoost classification model for intrusion detection in networks using GPU. *Electronics* 9(2):219
5. Card SK (2018) The psychology of human–computer interaction. CRC Press, Boca Raton
6. Card SK, Moran TP (1983) Newell, the psychology of human–computer interaction
7. Chen L, Fu J, Wu Y, Li H, Zheng B (2020) Hand gesture recognition using compact CNN via surface electromyography signals. *Sensors* 20(3):672
8. Chhabria K, Priya V, Thaseen IS (2020) Gesture recognition using deep learning. In: 2020 International conference on emerging trends in information technology and engineering (ic-ETITE). IEEE, pp 1–4
9. Clayton N, Emery N (2005) Corvid cognition. *Curr Biol* 15(3):R80–R81
10. Devineau G, Moutarde F, Xi W, Yang J (2018) Deep learning for hand gesture recognition on skeletal data. In: 2018 13th IEEE International conference on automatic face and gesture recognition (FG 2018). IEEE, pp 106–113
11. Ebeid IA, Zhang Y (2019) A systematic review of the literature in nature on human–computer interaction: preliminary results. *iConference 2019 proceedings*
12. Gadekallu TR, Rajput DS, Reddy MPK, Lakshman K, Bhattacharya S, Singh S, Jolfaei A, Alazab M (2020) A novel PCA-whale optimization-based deep neural network model for classification of tomato plant diseases using GPU. *J Real-Time Image Process*, 1–14
13. Garg S, Kaur K, Kumar N, Rodrigues JJ (2019) Hybrid deep-learning-based anomaly detection scheme for suspicious flow detection in SDN: a social multimedia perspective. *IEEE Trans Multimed* 21(3):566–578
14. Hu B, Wang J (2020) Deep learning based hand gesture recognition and UAV flight controls. *Int J Autom Comput* 17(1):17–29
15. Hussien AG, Amin M, Wang M, Liang G, Alsanad A, Gumaei A, Chen H (2020) Crow search algorithm: theory, recent advances, and applications. *IEEE Access* 8:173548–173565
16. Iwendi C, Uddin M, Ansere JA, Nkurunziza P, Anajemba JH, Bashir AK (2018) On detection of Sybil attack in large-scale VANETs using spider-monkey technique. *IEEE Access* 6:47258–47267
17. Javed AR, Usman M, Rehman SU, Khan MU, Haghighi MS (2020) Anomaly detection in automated vehicles using multistage attention-based convolutional neural network. *IEEE Trans Intell Transp Systems*
18. Jiang D, Li G, Sun Y, Kong J, Tao B (2019) Gesture recognition based on skeletonization algorithm and CNN with ASL database. *Multimed Tools Appl* 78(21):29953–29970
19. Jindal A, Aujla GS, Kumar N, Chaudhary R, Obaidat MS, You I (2018) Sedative: SDN-enabled deep learning architecture for network traffic control in vehicular cyber-physical systems. *IEEE Netw* 32(6):66–73
20. Kaggle: Hand gesture recognition database (2018). <https://www.kaggle.com/gti-upm/leapgestrecog/version/1> (Accessed 15 June 2020)
21. Karam M et al (2005) A taxonomy of gestures in human computer interactions
22. Köpklü O, Gunduz A, Kose N, Rigoll G (2020) Online dynamic hand gesture recognition including efficiency analysis. *IEEE Trans Biom Behav Identity Sci* 2(2):85–97
23. Kumar P, Tripathy B (2009) MMER: an algorithm for clustering heterogeneous data using rough set theory. *Int J Rapid Manuf* 1(2):189–207
24. Li G, Tang H, Sun Y, Kong J, Jiang G, Jiang D, Tao B, Xu S, Liu H (2019) Hand gesture recognition based on convolution neural network. *Cluster Comput* 22(2):2719–2729
25. Long M, Zeng Y (2019) Detecting iris liveness with batch normalized convolutional neural network. *Comput Mater Contin* 58(2):493–504
26. Luo Y, Qin J, Xiang X, Tan Y, Liu Q, Xiang L (2020) Coverless real-time image information hiding based on image block matching and dense convolutional network. *J Real-Time Image Process* 17(1):125–135
27. Maddikunta PKR, Gadekallu TR, Kaluri R, Srivastava G, Parizi RM, Khan MS (2020) Green communication in IOT networks using a hybrid optimization algorithm. *Comput Commun*
28. Mullick K, Namboodiri AM (2017) Learning deep and compact models for gesture recognition. In: 2017 IEEE international conference on image processing (ICIP. IEEE), pp 3998–4002
29. Nasreddine K, Benzinou A (2019) Shape geodesics for robust sign language recognition. *IET Image Process* 13(5):825–832
30. Oudah M, Al-Naji A, Chahl J (2020) Hand gesture recognition based on computer vision: a review of techniques. *J Imaging* 6(8):73
31. Pan L, Qin J, Chen H, Xiang X, Li C, Chen R (2019) Image augmentation-based food recognition with convolutional neural networks. *Comput Mater Contin* 59(1):297–313
32. Peng F, Zhou DL, Long M, Sun Xm (2017) Discrimination of natural images and computer generated graphics based on multi-fractal and regression analysis. *AEU-Int J Electron Commun* 71:72–81
33. Pinto RF, Borges CD, Almeida A, Paula IC (2019) Static hand gesture recognition based on convolutional neural networks. *J Electr Comput Eng*
34. Prior H, Schwarz A, Güntürkün O (2008) Mirror-induced behavior in the magpie (*pica pica*): evidence of self-recognition. *PLoS Biol* 6(8):e202
35. Qi J, Jiang G, Li G, Sun Y, Tao B (2020) Surface EMG hand gesture recognition system based on PCA and GRNN. *Neural Comput Appl* 32(10):6343–6351
36. Reddy GT, Khare N (2017) Hybrid firefly-bat optimized fuzzy artificial neural network based classifier for diabetes diagnosis. *Int J Intell Eng Syst* 10(4):18–27
37. Reddy GT, Reddy MPK, Lakshman K, Kaluri R, Rajput DS, Srivastava G, Baker T (2020) Analysis of dimensionality reduction techniques on big data. *IEEE Access* 8:54776–54788
38. Reddy T, Bhattacharya S, Maddikunta PKR, Hakak S, Khan WZ, Bashir AK, Jolfaei A, Tariq U (2020) Antlion re-sampling based deep neural network model for classification of imbalanced multimodal stroke dataset. *Multimed Tools Appl*, 1–25
39. Rehman ZU, Zia MS, Bojja GR, Yaqub M, Jinchao F, Arshid K (2020) Texture based localization of a brain tumor from MR-images by using a machine learning approach. *Med. Hypotheses*, p 109705
40. Rincon P (2005) Science/nature | crows and jays top bird IQ scale. BBC News
41. RM SP, Bhattacharya S, Maddikunta PKR, Somayaji SRK, Lakshman K, Kaluri R, Hussien A, Gadekallu TR (2020) Load balancing of energy cloud using wind driven and firefly algorithms in internet of everything. *J Parallel Distribut Comput*
42. RM SP, Maddikunta PKR, Parimala M, Koppu S, Reddy T, Chowdhary CL, Alazab M (2020) An effective feature engineering for DNN using hybrid PCA-GWO for intrusion detection in IOMT architecture. *Comput Commun*
43. Shen Y, Li J, Zhu Z, Cao W, Song Y (2015) Image reconstruction algorithm from compressed sensing measurements by dictionary learning. *Neurocomputing* 151:1153–1162
44. Shin HK, Ahn YH, Lee SH, Kim HY (2019) Digital vision based concrete compressive strength evaluating model using deep convolutional neural network. *CMC-Comput Mater Contin* 61(3):911–928

45. Skaria S, Al-Hourani A, Lech M, Evans RJ (2019) Hand-gesture recognition using two-antenna Doppler radar with deep convolutional neural networks. *IEEE Sens J* 19(8):3041–3048
46. Srivastava G, Deepa N, Prabadevi B, Reddy MPK (2021) An ensemble model for intrusion detection in the internet of software things. In: Adjunct proceedings of the 2021 international conference on distributed computing and networking, pp 25–30
47. Tan M, Zhou J, Xu K, Peng Z, Ma Z (2020) Static hand gesture recognition with electromagnetic scattered field via complex attention convolutional neural network. *IEEE Antennas Wirel Propag Lett* 19(4):705–709
48. Tripathy B, Mittal D (2016) Hadoop based uncertain possibilistic kernelized c-means algorithms for image segmentation and a comparative analysis. *Appl Soft Comput* 46:886–923
49. Vasan D, Alazab M, Wassan S, Naeem H, Safaei B, Zheng Q (2020) IMCFN: image-based malware classification using fine-tuned convolutional neural network architecture. *Comput Netw* 171:107138
50. Wang N, He M, Sun J, Wang H, Zhou L, Chu C, Chen L (2019) IA-PNCC: noise processing method for underwater target recognition convolutional neural network. *Comput Mater Contin* 58(1):169–181
51. Wei W, Dai Q, Wong Y, Hu Y, Kankanhalli M, Geng W (2019) Surface-electromyography-based gesture recognition by multi-view deep learning. *IEEE Trans Biomed Eng* 66(10):2964–2973
52. Yang XS (2011) Metaheuristic optimization. *Scholarpedia* 6(8):11472
53. Zehra W, Javed AR, Jalil Z, Khan HU, Gadekallu TR (2021) Cross corpus multi-lingual speech emotion recognition using ensemble learning. *Complex Intell Syst*, 1–10
54. Zhou L, Ma K, Wang L, Chen Y, Tang Y (2019) Binaural sound source localization based on convolutional neural network. *CMC Comput Mater Contin* 60(2):545–557

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.