

Human-Animal Affective Robot Touch Classification Using Deep Neural Network

Mohammed Ibrahim Ahmed Al-mashhadani¹, Theyazn H. H. Aldhyani^{2,*}, Mosleh Hmoud Al-Adhaileh³, Alwi M. Bamhdi⁴, Mohammed Y. Alzahrani⁵, Fawaz Waselallah Alsaade⁶ and Hasan Alkahtani^{1,6}

¹AL-Iraqia University, College of Education, Computer Department, Baghdad, Iraq

²Community College of Abqaiq, King Faisal University, Al-Ahsa, Saudi Arabia

³Deanship of E-learning and Distance Education, King Faisal University, Al-Ahsa, Saudi Arabia

⁴UMM Al-Qura University, College of Computing, Makkah, Saudi Arabia

⁵Department of Computer Sciences and Information Technology, Albaha University, Al Baha, Saudi Arabia

⁶College of Computer Science and Information Technology, King Faisal University, Al-Ahsa, Saudi Arabia

*Corresponding Author: Theyazn H. H. Aldhyani. Email: taldhyani@kfu.edu.sa

Received: 31 October 2020; Accepted: 25 January 2021

Abstract: Touch gesture recognition is an important aspect in human–robot interaction, as it makes such interaction effective and realistic. The novelty of this study is the development of a system that recognizes human–animal affective robot touch (HAART) using a deep learning algorithm. The proposed system was used for touch gesture recognition based on a dataset provided by the Recognition of the Touch Gestures Challenge 2015. The dataset was tested with numerous subjects performing different HAART gestures; each touch was performed on a robotic animal covered by a pressure sensor skin. A convolutional neural network algorithm is proposed to implement the touch recognition system from row inputs of the sensor devices. The leave-one-subject-out cross-validation method was used to validate and evaluate the proposed system. A comparative analysis between the results of the proposed system and the state-of-the-art performance is presented. Findings show that the proposed system could recognize the gestures in almost real time (after acquiring the minimum number of frames). According to the results of the leave-one-subject-out cross-validation method, the proposed algorithm could achieve a classification accuracy of 83.2%. It was also superior compared with existing systems in terms of classification ratio, touch recognition time, and data preprocessing on the same dataset. Therefore, the proposed system can be used in a wide range of real applications, such as image recognition, natural language recognition, and video clip classification.

Keywords: Touch gesture recognition; touch gesture classification; deep learning

1 Introduction

Social touch is a common method used to communicate interpersonal emotions. Social touch classification is an important research field that has great potential for improvement [1,2]. Social touch classification can be beneficial in many scientific applications, such as robotics and human–robot



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

interaction (HRI). A simple yet demanding question in this area is how to identify the type (or class) of a touch that affects a robot by analyzing the social touch gesture [3]. Each person can interact with their environment and other persons via touch sensors spread over humans. These touch sensors provide us with important information about objects that we handle, such as size, shape, position, surface, and movement. Touch is the simplest and most straightforward of all the sensors in the human body. Therefore, the touch system plays a main role in human life from its early stages [4]. Touch gestures are also crucial for human relationships [5,6].

The essential purpose of nonverbal (touch) interaction is to communicate and transfer emotions between humans. Thus, social touch is sometimes used to express the human state or to enable interaction between humans and animals or robots [7,8]. Social touch is also used to express different emotions in daily life, such as one's feeling upon accidentally bumping into a stranger in a busy store [9,10]. People use touch as a powerful method for social interaction. Through touch, people can express many positive and negative things, such as intent, understanding, affection, care, support, comfort, and (dis)agreement [11,12]. Different types of touch give various messages. For instance, a handshake is used as a greeting, whereas a slap is a punishment; petting is a calming gesture for both the person doing the petting and the animal being petted, and it reduces stress levels and evokes social responses from people [13–15]. This ability of humans to transfer significant emotions through touch language can be applied to robots by using artificial skin equipped with sensors [16,17]. The study of social touch recognition depends on the idea of the human ability to communicate emotions between them via touch [18]. Understanding how humans can elicit significant information from social touch helps designers develop algorithms and methods to simulate that response on robots in the correct form when they interact with humans [19]. Gesture patterns must be recognized in the correct form to help robots interpret and understand human gestures through interaction with them. Precise recognition enables robots to respond to humans and express their internal state and artificial emotions in a positive action. To ensure a high ratio of recognition accuracy, sensor devices must measure the touch pressure at high spatial and pressure resolutions [20]. Therefore, robots must be equipped with sensor devices that can detect emotions and facial expressions similar to human behavior [21]. The factors affecting touch interpretation include the relationships between people, cultures of people, the location of touch on the body, and the duration of the touch. Therefore, designers of artificial skin for robot bodies must take this into consideration to ensure that each touch conveys the real meaning that an individual intends to send to another [22,23]. Touch recognition systems can be developed through two methods: touch-pattern-based design (top-down approach) and touch-receptor-based design (bottom-up approach) [24]. In therapeutic and companionship applications between humans and pets or humanoid robots, closed-loop response can be prepared for social HRI [25,26]. This requires complex touch sensors and efficient interpretation [27]. Acceptable information about the operation of effective touch, such as possibilities and mechanisms, is needed [1,28]. Critical requirements, such as reliable control, perception, learning, and response in correct emotions, should be fulfilled to enable robots to participate in society and interact with humans in an effective manner [29]. The interaction behaviors of a real-world objects require the use of touch sensing for understanding humanoid robots [30,31]. The most critical issue in the design of social robots is how they can be taught to interact with users, store past participations, and use this stored information when responding to human touch [32,33]. A human can easily distinguish and understand touch gestures. However, in the HRI domain, an interface for recording social touch should first be developed. Researchers have proposed a setup for measuring touch pressure to record data and cover many social touch gesture classes. This setup uses a kind of artificial skin that records the pressure on it. The human–animal affective robot touch (HAART) dataset consists of seven touch gestures (pat, constant contact without movement [press], rub, scratch, stroke, tickle, and no touch) [34].

To establish a relationship between the required knowledge for objects and movements of the human hand, Lederman et al. [35] performed two experiments using haptic object exploration. The first experiment depended on a match-to-sample task, in which a directed match exists between the subject and a particular dimension. For exploring the object, “exploratory procedures” were used to classify hand movements. Each procedure had properties that were used by a matching process. The second experiment identified the reasons for special links that connected the exploratory procedures with the knowledge goals. During hand movement, the procedures were considered in terms of their necessity, sufficiency, and optimality of performance for each task. The obtained results had explained that through free exploration. The procedure is generally used to extract information about object properties because it is optimal, and sometimes even necessary, for these tasks. Reed et al. [36] attempted to recognize hand gestures in real time using a hidden Markov model (HMM) algorithm. The gesture recognition is based on the global features that are extracted from image sequences of a hand motion image database. The database contained 336 images showing dynamic hand gestures, such as hand waving, spinning, pointing, and moving. These gestures were performed by 14 participants, with each one performing 24 distinct gestures. The dataset was split to 312 samples for training and 24 samples for testing. The dynamic feature extraction reduced the amount of data by 0.3 of the original data information. The system achieved a gesture recognition accuracy of 92.2%. Colgan et al. [37] used video clips to study the reaction of 9–12-month-old infants with autism to different gestures. This study introduced the interaction of children who have autism with different types of social gestures that develop their nonverbal communication skills. Three types of touch functions were used in this study: joint attention, behavior regulation, and social interaction. These touch functions contain diverse gestures. Joint attention gestures refer to paying attention to the body. Behavior regulation gestures are gestures that can control the behavior of another gesture. Social interaction gestures refer to gestures that are used for social interaction with other humans.

Haans et al. [38] explained certain issues related to mediated social touch. These issues include perceptual mechanisms, enabling technologies, theoretical underpinnings, and methods or algorithms used to solve the problems of artificial skin.

Fang et al. [39] proposed a new method that depends on hand gesture recognition for interaction between humans and computers in real time. The hand is represented in multiple gestures by using elastic graphs with local jets of a Gabor filter used for feature extraction. Users perform hand gestures to recognize the gestures. The proposed method has three steps. First, a hand image is segmented into color and motion cues generated by detection and tracking. Second, features are extracted by a scale space. The last step is the hand gesture recognition. The recognition ratio is effected by camera movement in virtue of stable hand tracking. A boosted classifier tree was used to recognize the following hand gestures: left, right, up, down, open, and close. A total of 2596 frames were recorded in the experiment; 2436 frames were recognized correctly, and the correct classification ratio was 93.8%.

Jia et al. [40] described the design and implementation of an intelligent wheelchair (IW). The motion of this wheelchair is controlled via recognition of head gestures based on HRI. To detect faces correctly in real time, the authors used a hybrid method that combines camshaft object tracking with a face detection algorithm. In addition, the interface between the user and the wheelchair is equipped with traditional control tools (joystick, keyboard, mouse, and touchscreen), voice-based control (audio), vision-based control (cameras), and other sensor-based control (infrared sensors, sonar sensors, pressure sensors, etc.).

Therefore, our aim in this paper is to classify touch in (almost) real time and determine how much data (how many frames) on average would be needed to classify touch gestures. Our method can lead to more realistic and near-real-time HRI. In this paper, we propose a model that addresses the touch gesture classification problem without preprocessing. The use of sensor data without preprocessing is a powerful

approach to efficiently classifying required gesture types. To be able to handle this large amount of data, we use an effective tool that widely explored in the literature. Deep learning is based on artificial neural networks (ANNs), which are currently surpassing classical methods in performance, especially in pattern recognition fields. For example, Google's use of deep learning leads to a CCR of more than 80% in 487 classes of videos where the input dimension is $170 \times 170 \times 3 \times N$. The key points of the proposed algorithm are as follows.

1. It achieves high accuracy and outperforms other classification algorithms on the same dataset.
2. It uses a convolutional neural network (CNN) to classify touch gestures in an end-to-end architecture.
3. It predicts the class of a touch gesture in almost real time.
4. It can start classification operation after the minimum number of frames is received.
5. The system can classify gestures even though the training data is considered in the middle of the gesture.

2 Materials and Methods

In this section, the proposed recognition model, namely, the HAART recognition model, is explained. A CNN was implemented to develop the system.

2.1 HAART Dataset

The HAART, the dataset was collected from 10 participants. Each one performed 7 different touch gestures: pat, constant contact without movement (press), rub, scratch, stroke, tickle, and no touch [28]. The touch was recorded on a 10×10 pressure sensor. The duration of each touch was 10 s, and the data were sampled at 54 Hz. The sensor was wrapped around a robotic animal model. The permutations of 4 cover conditions (none, short minky, long minky, and synthetic fur) and 3 substrate conditions (firm and flat, foam and flat, and foam and curve). Therefore, the data initially included 840 gestures (7 gestures \times 10 participants \times 12 conditions). However, due to technical problems, 11 gestures were lost from the dataset, so 829 gestures remained. The data were minimized to include a middle 8×8 sensor grid and middle 8 s. This led to 432 frames for each participant, gesture, substrate, and cover condition [41].

2.2 Deep Learning Algorithm

Deep neural networks are ANNs with multiple layers. In the last decades, ANNs have been considered effective algorithms for handling real-time applications [42]. Deep learning algorithms use many deep hidden layers, thus surpassing classical ANN methods [43,44]. CNNs are a widely known type of deep neural network algorithms; they are named such because they use linear mathematical operations between matrices.

2.2.1 Convolutional Neural Network (CNN)

CNN algorithms have yielded groundbreaking results during the past decade in various applications, such as pattern recognition, computer vision, voice recognition, and text mining. The advantage of CNNs is that they reduce the number of parameters that are required in ANN algorithms [45]. This improvement has compelled researchers to use CNNs to develop systems. The most significant advantage of applying CNN algorithms is producing features that are not spatially dependent [46]. The convolutional layer will identify the number and size of the receptive field of neurons in the layer (L), which is connected to a single neuron in the next layer, using a scalar product between their weights and the region connected to the input volume. In the convolutional layer, given the input, a weight matrix all the input is being passed over, and the recorded weighted summation is placed as a single element of the subsequent layer. Three hyperparameters, namely, filter size, stride, and zero padding, affect the performance of the convolutional layer. By using different values for these hyperparameters, the convolutional layer can decrease the complexity of the network. A CNN algorithm has different layers. Fig. 1 shows the details of CNN algorithm layers.

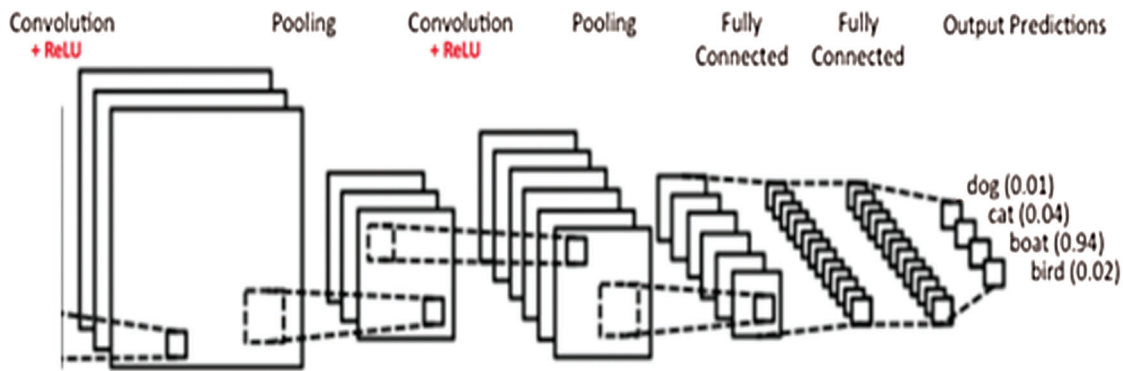


Figure 1: CNN layers

2.2.1.1 Non-linear Layer

Non-linearity can be used to adjust or cut off the generated output. There are many nonlinear functions that can be used in CNN. However, Rectified Linear Unit ReLU is one of the most common nonlinear functions applied in image processing applications. It is shown in Fig. 2. The main goal of using the ReLU is applying an element-wise activation function to the feature map from the previous layer. In addition, the ReLU function transfers all value of the features map to positive or zero. It can be represented as shown in Eq. (1).

$$ReLU(x) = [0 \text{ if } x < 0, x \text{ if } x \geq 0] \quad (1)$$

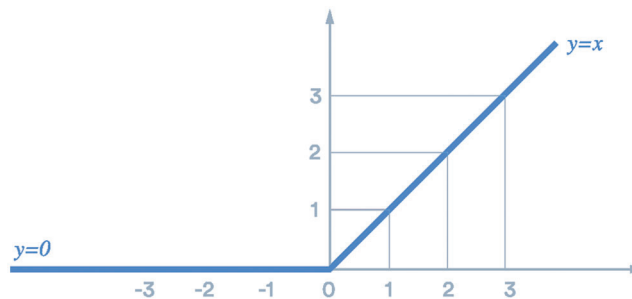


Figure 2: ReLU function

2.2.1.2 Pooling Layer

A pooling layer roughly reduces the dimensions of the input data and minimizes the number of parameters in the feature map. The simplest way to implement a pooling layer is by selecting the maximum of each region and then writing it in the corresponding place of the next layer. The use of this pooling filter reduces the input size to 25% of its original size. Averaging is another pooling method, but the selection of the maximum is the more widely used method in the literature. The maximum pooling method is non-invertible, so the original values (that is, the values before the pooling operation) cannot be restored. Nonetheless, the original values can be approximated by recording the locations of the maximum values of each moving in a set of switch variables.

2.2.1.3 Softmax Layer

The softmax function (sometimes called normalized exponential function) is considered the best method of showing the categorical distribution. The input of the softmax function is an N-dimensional vector of

units, and each unit is expressed by an arbitrary real value, whereas the output is an M-dimensional vector ($N \mu M$) with real values ranging between 0 and 1. A large value is changed to a real number close to one, and a small value is changed to a real number close to zero. The summation of all output values must be equal to 1. Therefore, an output with a large probability is unchanged. The softmax function is used to calculate the probability distribution of an N-dimensional vector. In general, softmax is used at the output layer for multiclass classification in machine learning, deep learning, and data science. Correct calculation of the output probability helps determine the proper target class for the input dataset. The probabilities of the maximum values are increased by using an exponential element. The softmax equation is shown by Eq. (2)

$$O_i = \frac{e^{z_i}}{\sum_{i=1}^M e^{z_i}} \quad (2)$$

where i , z_i is the output, O_i indicates the softmax output, and M is the total number of output nodes. Fig. 3 demonstrates the softmax layer in the network

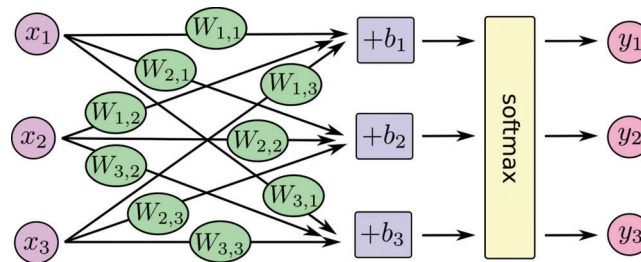


Figure 3: Softmax layer in network

2.2.1.4 Fully-Connected Layer

The fully connected layer is the last layer in any CNN. Each node in the layer L is connected directly to each node in layers $L - 1$ and $L + 1$. There is no connection between nodes in the same layer, in contrast to the traditional ANN. Therefore, this layer requires a long training and testing time. More than one fully connected layer can be used in the same network, as shown in Fig. 4.

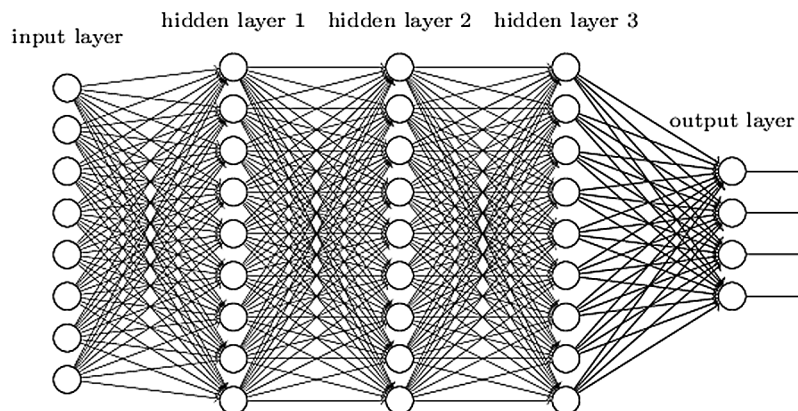


Figure 4: Fully connected layer of CNN

2.3 Proposed Model

The HAART dataset, which was recorded on a pressure sensor, has a size of 8×8 , and the data length is equal to N . Thus, the data are in three dimensions. The data are sampled at 54 Hz, so the value of N for each gesture is $54 \times 8 = 432$. The input to the CNN must be of equal size (images or frames). Therefore, the raw sensor data are used as input to the network for touch gesture classification. The main challenge of this research is to discover the significant architecture of CNN algorithms for gesture recognition. Hence, first, we define the input and the output structure of the network. Then, through some experiments, we present the best architecture according to the obtained results. Each recorded sample is an $8 \times 8 \times N$ matrix. An input of $8 \times 8 \times 432$ to the CNN will be computationally intensive. Therefore, we split each sample into subsamples with a fixed length. The optimal frame length (F) will be determined by the results.

Consequently, the sample at our disposal can be broken down into 86 multiple subsamples of $8 \times 8 \times 5$ with the same label (e.g., message). The resulting sample has a $8 \times 8 \times \text{frame length } (F)$, as shown in Eq. (3). Performance is improved by using part of the sample.

$$F = \text{floor} \left(\frac{N}{L} \right) = \text{floor} \left(\frac{432}{86} \right) = 5 \quad (3)$$

where F is the frame length, $N = 432$, and $L = 5, 10, 15$. In this case, the first subsample is picked, and the others are kept away. For example, from the $8 \times 8 \times 432$ samples, we will have only $8 \times 8 \times 43$ (if $L = 10$), which is the first part of the big sample. This can be useful, given that most of the gestures differ considerably only in the beginning of performing them, not in the middle of performing. The second reason is that we will have an equal number of samples from each class. Splitting samples leads to having more samples in the gestures with time. In summary, the idea is to use raw data for the classification; that is, for each experiment, we will have a recording with a length of $8 \times 8 \times F$, which will be fed to the deep neural network. Thus, the procedure can be regarded as similar to video classification. However, the frame length (F) differs for each experiment from 1 (when $L = 432$) to 432 (when $L = 1$). To fix this problem, we test the performance of the network for different values of L . This idea has the following benefits.

- I Based on the frame length, the number of samples for training the neural network can be increased. The subsamples can be gained through short frame lengths with less information in each subsample, and vice versa.
- II Subsamples can be obtained from different parts of the main sample. In other words, the adopted method helps in recognizing the touch gesture in the middle or the end of performing it.
- III Earlier studies were not designed for real-time classification. Recognition of a gesture class requires waiting until the touch gesture is fully performed. By contrast, the adopted approach in this study allows recognition of the gesture after receiving a certain length of data.

The shape of the output in the adopted method is a softmax function with 7 classes. Although the highest value in the output node is used, the values of the softmax function can be used to explore other highly probable hypotheses. Our touch gesture recognition input is in the form of $8 \times 8 \times F$, where F is the number of filters. More channels can be obtained by increasing the frame length. This will result in more convolutions, which will be more computationally intensive. Convolutional layers can be cascaded together to build the classifier. Each convolutional network comprises a convolutional layer, nonlinear layer, pooling layer, one fully connected layer, and softmax, which are considered for gesture recognition. Tab. 1 shows of parameters of the CNN algorithm.

Table 1: Significant parameters of CNN model

Layer No.	Elements	Parameters	Values
1	Convolutional Filter	Input Channels	$8 \times 8 \times 14$
		Size	2×2
		Stride	1
		Pad	1
	Max-Pooling	Size	2×2
		Pad	2
2	Convolutional Filter	Input Channels	32
		Size	2×2
		Stride	1
		Pad	1
	Max-Pooling	Size	2×2
		Pad	2
3	Convolutional Filter	Input Channels	64
		Size	3×3
		Stride	1
		Pad	1
	Max-Pooling	Size	2×2
		Pad	2
4	Fully-Connected	Input to layer	$128 \times 2 \times 2$
		Output from Layer	256
5	Softmax	Output Units	7

3 Results and Discussion

A grid search is performed to select the optimal frame length (F). We run experiments for $L = 5, 10, 15, \dots, 50$ to determine the most accurate classification based on leave-one-subject-out cross-validation. Since these experiments are computationally expensive, we select only ID = 1, 3, 5, 7 to select the value of L for the cross-validation. The average cross-validation accuracy is the criterion for the optimal frame length. The results are shown in Fig. 5 for each selected subject and its average. Overall, a long frame length is beneficial for the classification rate. As the value of L increases, the performance of the network increases; it reaches $L = 30$, which indicates maximum performance. However, the performance of the network begins to decline as the value of L increases further. Therefore, among the selected frame lengths, $L = 30$ provides the maximum classification rate; hence, it is selected as the input dimension of the CNN.

The leave-one-subject-out cross-validation results of the proposed system on all the subjects are presented in Tab. 2. The accuracy of the classification ratio is 83.2%, which is 11.8% better than the state-of-the-art result.

For a better understanding of the results, a confusion matrix is presented in Tab. 3. In the table, there are a few large no diagonal numbers, which show the biggest confusion in the proposed system. Mutual confusion is

observed in the following classes: Pat and Rub, Pat and Stroke, Scratch and Rub, Tickle and Scratch, and Tickle and Rub. Such confusion is natural because these movements are performed similarly by humans.

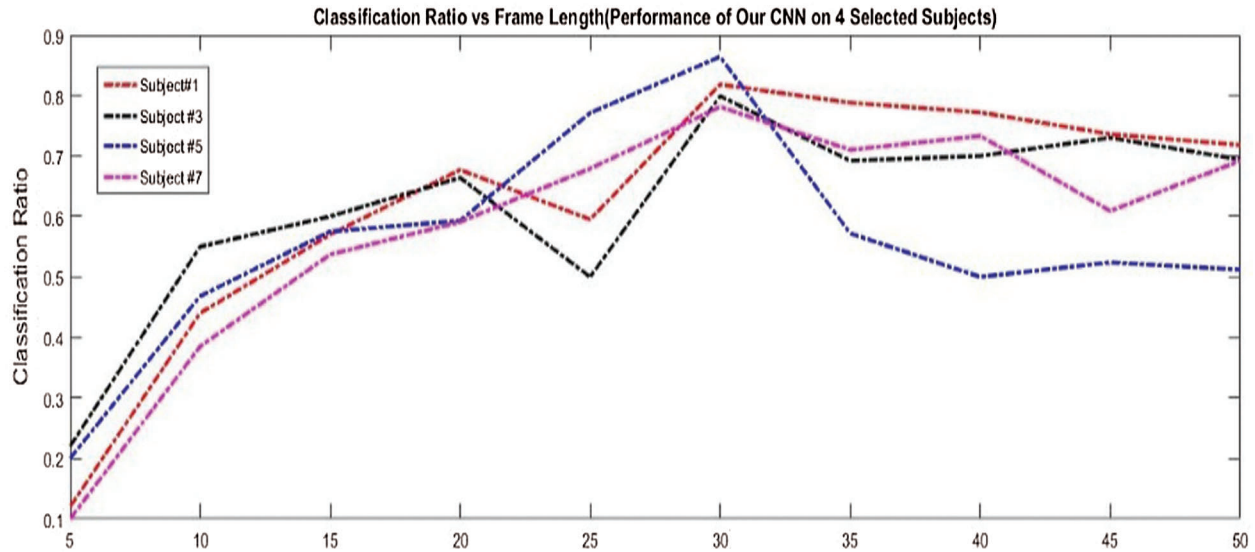


Figure 5: Performance of CNN algorithm with three convolutional layers over four randomly selected subjects on HAART dataset

Table 2: Performance evaluation of proposed system using leave-one-subject-out cross-validation on all subjects

Validation Method	Classification Ratio	Standard Deviation
One-Subject-Leave-Out	83.2%	13%

Table 3: Confusion matrix of proposed system for gesture recognition

Gesture	a	b	c	d	e	f	g
Constant (a)	35	0	0	0	0	0	0
No touch (b)	1	36	0	0	0	0	0
Pat (c)	0	0	29	1	0	0	0
Rub (d)	0	0	3	30	7	3	2
Scratch (e)	0	0	0	2	26	0	20
Stroke (f)	0	0	4	2	0	33	1
Tickle (g)	0	0	0	1	3	0	13

The results of the proposed system slightly differ due to the touch gesture class. Fig. 6 demonstrates the performance of the proposed system using different touch gesture classes. In the proposed system, we use the Tickle and Scratch touch gesture class, which presents multiple mutual conflicts with other classes. The proposed system performs the best in the Constant and No Touch classes. For a comparison of the

proposed system with other classification methods applied on the same dataset, [Tab. 4](#) shows that our proposed system improves the classification ratio without preprocessing. Moreover, it depends on the original input data, not on feature extraction, where some information from the raw data is lost.

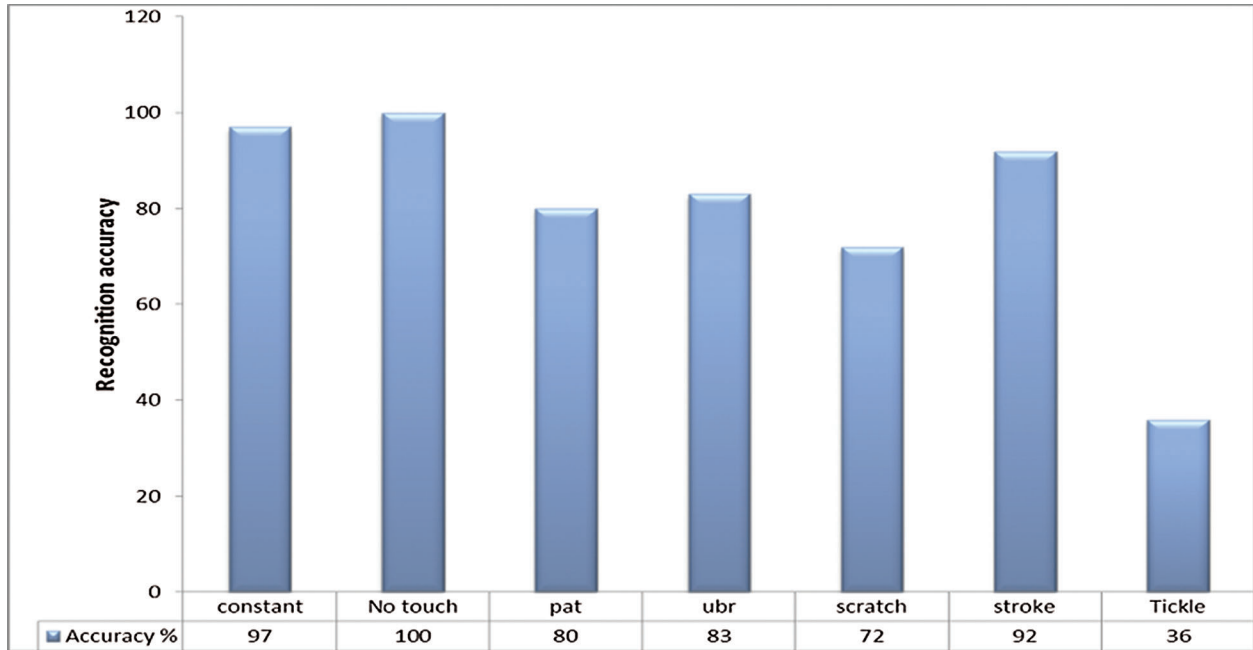


Figure 6: Performance of proposed system

Table 4: Comparison results of proposed system against existing classification using HAART dataset

References	Preprocessing	# Features	Classification Method	Accuracy (%)
Balli Altuglu et al. [47]	Yes	42	Random Forest (RF)	61.5
Gaus et al. [48]	Yes	5	Random Forest (RF)	66.53
			Boosting	64.54
Hughes et al. [49]	Yes	7	Deep Autoencoders	71.4
Ta et al. [50]	Yes	273	SVM	68.52
			Random Forest (RF)	70.91
Hughes et al. [51]	Yes	7	CNN	56.1
			CNN-RNN	61.35
			Autoencoder- RNN	61.35
Proposed System	No	Input Data (Raw Data)	CNN	83.2

4 Conclusion

We propose a system for classifying touch gestures using a deep neural network. The CNN is selected because it is a good approach to feature extraction. The HAART dataset is selected to train the CNN due to the variety of classes. The proposed system yields an accuracy of 83.2%. A comparative classification of the

results between the proposed system and state-of-the-art systems is presented. The findings indicate that the proposed system achieves successful results. There are two benefits of the proposed system compared with the existing systems in the literature. First, the proposed system does not require preprocessing or manual feature extraction approaches, and it can be implemented end to end. Second, the proposed system can recognize the class once the minimum number of frames is received. This minimum number of frames is found in the HAART dataset using a grid search. However, the size of the input frame (8×8) negatively affects the CNN's performance, as the CNN reduces the size of the frame when being transferred from one layer to another. Finally, the proposed system outperforms the compared state-of-the-art systems on the HAART dataset. Sparse-coding-based methods, autoencoder-based methods, and restricted Boltzmann machines can be used to further improve the developed system.

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] A. Flagg and K. MacLean, "Affective touch gesture recognition for a furry zoomorphic machine," in *Proc. of the 7th Int. Conf. on Tangible, Embedded and Embodied Interaction*, New York, NY, USA, pp. 25–32, 2019.
- [2] F. R. Ortega, N. Rische, A. Barreto, F. Abyarjoo and M. Adjouadi, "Multi-touch gesture recognition using feature extraction," in *Innovations and Advances in Computing, Informatics, Systems Sciences, Networking and Engineering*. Switzerland, pp. 291–296, 2015.
- [3] J. Chang, K. MacLean and S. Yohanan, "Gesture recognition in the haptic creature," in *Haptics: Generating and Perceiving Tangible Sensations*. Berlin, Heidelberg: Verlag, pp. 385–391, 2010.
- [4] T. H. H. Aldhyani, M. Al-Yaari, H. Alkahtani and M. Maashi, "Water quality prediction using artificial intelligence algorithms," *Applied Bionics and Biomechanics*, vol. 2020, pp. 12, 2020.
- [5] H. Alkahtani, T. H. H. Aldhyani and M. Al-Yaari, "Adaptive anomaly detection framework model objects in cyberspace," *Applied Bionics and Biomechanics*, vol. 2020, pp. 14, 2020.
- [6] M. M. Jung, L. V. D. Leij and S. M. Kelders, "An exploration of the benefits of an animallike robot companion with more advanced touch interaction capabilities for dementia care," *Frontiers in ICT*, vol. 4, pp. 16, 2017.
- [7] J. N. Bailenson, N. Yee, S. Brave, D. Merget and D. Koslow, "Virtual interpersonal touch: Expressing and recognizing emotions through haptic devices," *Human-Computer Interaction*, vol. 22, no. 3, pp. 325–353, 2007.
- [8] S. Kratz and M. Rohs, "A gesture recognizer: Simple gesture recognition for devices equipped with 3d acceleration sensors," in *Proc. of the 15th Int. Conf. on Intelligent User Interfaces*, Hong Kong, China, pp. 341–344, 2010.
- [9] G. Huisman, A. D. Frederiks, B. V. Dijk, D. Hevlen and B. Krose, "The TASST: Tactile sleeve for social touch," in *World Haptics Conf. (WHC)*, Canada, pp. 211–216, 2013.
- [10] X. Zhang, X. Chen, W. H. Wang, J. H. Yang, V. Lantz *et al.*, "Hand gesture recognition and virtual game control based on 3D accelerometer and EMG sensors," in *Proc. of the 14th Int. Conf. on Intelligent user interfaces*, Sanibel Island, Florida, USA, pp. 401–406, 2009.
- [11] A. Kotranza, B. Lok, C. M. Pugh and D. S. Lind, "Virtual humans that touch back: Enhancing nonverbal communication with virtual humans through bidirectional touch," in *Virtual Reality Conf.* Louisiana, USA, 175–178, 2009.
- [12] S. Yohanan, J. Hall, K. MacLean, E. Croft, M. D. Loos *et al.*, "Affect-driven emotional expression with the haptic creature," in *Proc. of UIST, User Interface Software and Technology*, Canada, pp. 2, 2009.
- [13] W. Stiehl and C. Breazeal, "Affective touch for robotic companions," *Affective Computing and Intelligent Interaction*, USA, pp. 747–754, 2005.

- [14] W. D. Stiehl, J. Lieberman, C. Breazeal, L. Basel, L. Lalla *et al.*, “Design of a therapeutic robotic companion for relational, affective touch,” in *IEEE Int. Workshop on Robot and Human Interactive Communication, 2005. ROMAN 2005*, pp. 408–415, 2005.
- [15] H. Knight, R. Toscano, W. D. Stiehl, A. Chang, Y. Wang *et al.*, “Real-time social touch gesture recognition for sensate robots,” in *Int. Conf. on Intelligent Robots and Systems*. Louis, MO, USA, pp. 3715–3720, 2009.
- [16] A. Flagg, D. Tam, K. MacLean and R. Flagg, “Conductive fur sensing for a gesture-aware furry robot,” in *Haptics Symposium (HAPTICS)*, Vancouver, BC, Canada: IEEE, pp. 99–104, 2012.
- [17] M. A. Hoepflinger, C. D. Remy, M. Hutter, L. Spinello and R. Siegwart, “Haptic terrain classification for legged robots,” in *2010 IEEE Int. Conf. on Robotics and Automation (ICRA)*. Anchorage, AK, USA, pp. 2828–2833, 2010.
- [18] A. F. Dugas, Y. H. Hsieh, S. R. Levin, J. M. Pines, D. P. Mareiniss *et al.*, “Google flu trends: Correlation with emergency department influenza rates and crowding metrics,” *Clinical Infectious Diseases*, vol. 54, no. 4, pp. 463–469, 2012.
- [19] D. S. Tawil, D. Rye and M. Velonaki, “Touch modality interpretation for an EIT-based sensitive skin,” in *IEEE Int. Conf. on Robotics and Automation (ICRA)*. Shanghai, China, pp. 3770–3776, 2011.
- [20] F. Naya, J. Yamato and K. Shinozawa, “Recognizing human touching behaviors using a haptic interface for a pet-robot,” in *IEEE Int. Conf. on Systems, Man, and Cybernetics IEEE SMC’99 Conf. Proc.*, Tokyo, Japan, pp. 1030–1034, 1999.
- [21] L. Cañamero and J. Fredslund, “I show you how I like you-can you read it in my face?,” *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 31, no. 4, pp. 454–459, 2001.
- [22] X. L. Cang, P. Bucci, A. Strang, J. Allen, K. MacLean *et al.*, “Different strokes and different folks: Economical dynamic surface sensing and affect-related touch recognition,” in *Proc. of the 2015 ACM on Int. Conf. on Multimodal Interaction*, pp. 147–154, 2015.
- [23] E. Kerruish, “Affective touch in social robots,” *Transformations*, vol. 29, no. 2, pp. 117–135, 2017.
- [24] Y. M. Kim, S. Y. Koo, J. G. Lim and D. S. Kwon, “A robust online touch pattern recognition for dynamic human-robot interaction,” *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1979–1987, 2010.
- [25] K. Altun and K. E. MacLean, “Recognizing affect in human touch of a robot,” *Pattern Recognition Letter*, vol. 66, pp. 31–40, 2015.
- [26] F. N. Newell, M. O. Ernst, B. S. Tjan and H. H. Bühlhoff, “Viewpoint dependence in visual and haptic object recognition,” *Psychological Science*, vol. 12, no. 1, pp. 37–42, 2001.
- [27] Q. Chen, N. D. Georganas and E. M. Petriu, “Real-time vision-based hand gesture recognition using haar-like features,” in *Instrumentation and Measurement Technology Conf. Proc.*, Warsaw, Poland: IEEE, pp. 1–6, 2007.
- [28] S. Yohanan and K. E. MacLean, “The role of affective touch in human-robot interaction: Human intent and expectations in touching the haptic creature,” *International Journal of Social Robotics*, vol. 4, no. 2, pp. 163–180, 2012.
- [29] U. Martinez-Hernandez and T. J. Prescott, “Expressive touch: Control of robot emotional expression by touch,” in *2016 25th IEEE Int. Symposium on Robot and Human Interactive Communication (RO-MAN)*, New York, NY, USA, pp. 974–979, 2016.
- [30] R. S. Dahiya, G. Metta, M. Valle and G. Sandini, “Tactile sensing—From humans to humanoids,” *IEEE Transactions on Robotics*, vol. 26, no. 1, pp. 1–20, 2010.
- [31] R. L. Klatzky and J. Peck, “Please touch: Object properties that invite touch,” *IEEE Transactions on Haptics*, vol. 5, no. 2, pp. 139–147, 2012.
- [32] U. Martinez-Hernandez, A. Damianou, D. Camilleri, L. W. Boorman, N. Lawrence *et al.*, “An integrated probabilistic framework for robot perception, learning and memory,” in *IEEE Int. Conf. on Robotics and Biomimetics (ROBIO)*. Qingdao, China, pp. 1796–1801, 2016.
- [33] M. J. Hertenstein, J. M. Verkamp, A. M. Kerestes and R. M. Holmes, “The communicative functions of touch in humans, nonhuman primates, and rats: A review and synthesis of the empirical research,” *Genetic, Social, and General Psychology Monographs*, vol. 132, no. 1, pp. 5–94, 2006.

- [34] X. L. Cang, P. Bucci, A. Strang, J. Allen, K. MacLean *et al.*, “Different strokes and different folks: Economical dynamic surface sensing and affect-related touch recognition,” in *Proc. of ACM on Int. Conf. on Multimodal Interaction*, pp. 147–154, 2015.
- [35] S. J. Lederman and R. L. Klatzky, “Hand movements: A window into haptic object recognition,” *Cognitive Psychology*, vol. 19, no. 3, pp. 342–368, 1987.
- [36] C. L. Reed, R. J. Caselli and M. J. Farah, “Tactile agnosia: Underlying impairment and implications for normal tactile object recognition,” *Brain*, vol. 119, no. 3, pp. 875–888, 1996.
- [37] S. E. Colgan, E. Lanter, C. McComish, L. R. Watson, E. R. Crais *et al.*, “Analysis of social interaction gestures in infants with autism,” *Child Neuropsychology*, vol. 12, no. 4–5, pp. 307–319, 2006.
- [38] A. Haans and W. Jsselsteijn, “Mediated social touch: A review of current research and future directions,” *Virtual Reality*, vol. 9, no. 2–3, pp. 149–159, 2006.
- [39] Y. Fang, K. Wang, J. Cheng and H. Lu, “A real-time hand gesture recognition method,” in *IEEE Int. Conf. on Multimedia and Expo*. Beijing, China, 995–998, 2007.
- [40] P. Jia, H. H. Hu, T. Lu and K. Yuan, “Head gesture recognition for hands-free control of an intelligent wheelchair,” *Industrial Robot: An International Journal*, vol. 34, pp. 60–68, 2007.
- [41] X. L. Cang, P. Bucci, A. Strang, J. Allen, K. MacLean *et al.*, “Different strokes and different folks: Economical dynamic surface sensing and affect-related touch recognition,” in *Proc. of the ACM on Int. Conf. on Multimodal Interaction*, Seattle Washington, USA, pp. 147–154, 2015.
- [42] T. H. Aldhyani, A. S. Alshebami and M. Y. Alzahrani, “Alzahrani Soft computing model to predict chronic diseases,” *Information Science and Engineering*, vol. 36, no. 2, pp. 365–376, 2020.
- [43] T. H. Aldhyani, A. S. Alshebami and M. Y. Alzahrani, “Soft clustering for enhancing the diagnosis of chronic diseases over machine learning algorithms,” *Healthcare Engineering*, vol. 2020, no. 4, pp. 16, 2020.
- [44] T. H. H. Aldhyani, M. Alrasheedi, A. A. Alqarni, M. Y. Alzahrani and A. M. Bamhdi, “Intelligent hybrid model to enhance time series models for predicting network traffic,” *IEEE Access*, vol. 8, pp. 130431–130451, 2020.
- [45] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu *et al.*, “Deep learning for visual understanding: A review,” *Neurocomputing*, vol. 187, no. 1, pp. 27–48, 2016.
- [46] K. He, X. Zhang, S. Ren and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” in *European Conf. on Computer Vision*. USA, pp. 346–361, 2014.
- [47] T. B. Altuglu and K. Altun, “Recognizing touch gestures for social human-robot interaction,” in *Proc. of the ACM on Int. Conf. on Multimodal Interaction*, Seattle Washington, USA, pp. 407–413, 2015.
- [48] Y. F. A. Gaus, T. Olugbade, A. Jan, R. Qin, J. Liu *et al.*, “Social touch gesture recognition using random forest and boosting on distinct feature sets,” in *Proc. of the ACM on Int. Conf. on Multimodal Interaction*, Seattle Washington, USA, pp. 399–406, 2015.
- [49] D. Hughes, N. Farrow, H. Profita and N. Correll, “Detecting and identifying tactile gestures using deep autoencoders, geometric moments and gesture level features,” in *Proc. of the 2015 ACM on Int. Conf. on Multimodal Interaction*, Seattle, Washington, USA, pp. 415–422, 2015.
- [50] V. C. Ta, W. Johal, M. Portaz, E. Castelli and D. Vaufraydaz, “The Grenoble system for the social touch challenge at ICMI 2015,” in *Proc. of the 2015 ACM on Int. Conf. on Multimodal Interaction*, Seattle Washington, USA, pp. 391–398, 2015.
- [51] D. Hughes, A. Krauthammer and N. Correll, “Recognizing social touch gestures using recurrent and convolutional neural networks,” in *Int. Conf. on Robotics and Automation (ICRA)*. Singapore, 2315–2321, 2017.