Wang *et al. Health Inf Sci Syst (2021) 9:30*
https://doi.org/10.1007/s13755-021-00159-3

Health Information Science and Systems

RESEARCH

Hand tremor detection in videos with cluttered background using neural network based approaches

Xinyi Wang[1,2*], Saurabh Garg[1], Son N. Tran[1], Quan Bai[1] and Jane Alty[2]

Abstract

With the increasing prevalence of neurodegenerative diseases, including Parkinson's disease, hand tremor detection has become a popular research topic because it helps with the diagnosis and tracking of disease progression. Conventional hand tremor detection algorithms involved wearable sensors. A non-invasive hand tremor detection algorithm using videos as input is desirable but the existing video-based algorithms are sensitive to environmental conditions. An algorithm, with the capability of detecting hand tremor from videos with a cluttered background, would allow the videos recorded in a non-research environment to be used. Clinicians and researchers could use videos collected from patients and participants in their own home environment or standard clinical settings. Neural network based machine learning architectures provide high accuracy classification results in related fields including hand gesture recognition and body movement detection systems. We thus investigated the accuracy of advanced neural network architectures to automatically detect hand tremor in videos with a cluttered background. We examined configurations with different sets of features and neural network based classification models. We compared the performance of different combinations of features and classification models and then selected the combination which provided the highest accuracy of hand tremor detection. We used cross validation to test the accuracy of the trained model predictions. The highest classification accuracy for automatically detecting tremor (vs non tremor) was 80.6% and this was obtained using Convolutional Neural Network-Long Short-Term Memory and features based on measures of frequency and amplitude change.

Keywords: Advanced neural network, Hand tremor detection, Machine learning, Videos with cluttered background

Introduction

Hand tremor detection is a popular research topic as it assists with the early detection and diagnosis of neurodegenerative diseases including Parkinson's disease and other neurological disorders. Parkinson's disease is the fastest growing neurological disorder worldwide [1]. The incidence of Parkinson's disease is estimated to be 1 in 500 of the adult population, with the incidence increasing to 1 in 100 for individuals aged over 65. Approximately 70% of people with Parkinson's disease have tremor [2] and there is approximately only 75% clinical diagnostic accuracy [3]. Other tremor disorders, such as Essential Tremor and drug-induced tremor, are also very common, and we lack clinically accessible methods to quantify the presence and severity of tremor. Therefore, a tremor detection system with accurate prediction ability would help clinicians and researchers alike.

Conventional methods used wearable sensors, such as accelerometers, to detect hand tremor [4–6]. This method is time consuming and requires a high level of professional input—to set up the sensors, collect data and interpret the findings. In addition, there are infection control issues with patients sharing the same sensors, which is particularly problematic in the COVID-19 pandemic. Researchers have also used Leap Motion Controller (a haptic device for hand tracking) to detect hand tremor, which in comparison with wearable sensors

*Correspondence: xinyi.wang@utas.edu.au
[2] Wicking Dementia Research and Education Centre, College of Health and Medicine, University of Tasmania, Hobart, TAS 7000, Australia
Full list of author information is available at the end of the article

© The Author(s), under exclusive licence to Springer Nature Switzerland AG 2021.

dramatically decreased the complexity of setting up the environment for data collection and analysis [7, 8]. However, there remains an urgent need to find new non-invasive and cheaper methods such as using videos to detect hand tremor [3, 9, 10]. Soran et al. [9] used Optical Flow and Support Vector Machine (SVM) to extract features of hand tremor from video and perform classification but, as a color histogram was used to detect the hand region, these methods only worked using a plain background. A non-plain, or 'cluttered background' means that there is movement or variation in color in the background. For a video-based tremor detection system to be useful in the real-world settings of clinics, research trials, and home-monitoring, it will need to be able to work with a cluttered background too.

In recent years, methods using advanced neural network architectures showed promising prediction accuracy in fields related to hand tremor detection, including hand gesture recognition and body movement detection systems [8, 11–13]. However, their performance in the context of hand tremor is still unexplored. Thus, our research objective is to investigate the applicability of new machine learning methods such as advanced neural network algorithms to accurately detect hand tremor from videos with a cluttered background. We investigate different neural network architectures including Long Short-term Memory (LSTM) architecture and Convolutional Neural Network-Long Short-term Memory (CNN-LSTM) architecture.

The proposed hand tremor detection algorithms contain a unique combination of three stages including automatic hand region detection, feature extraction and classification. To detect the hands in a cluttered background, a machine learning based automated approach named Mediapipe [14] is utilized. The hand region detection algorithm takes videos as input and outputs the normalized coordinates of 21 landmarks on a hand in every frame [15]. We investigated two set of extracted features: the first is the frequency of motion directional changes and the second is the change in distance of hand movement. These are measures of "tremor frequency" and "tremor amplitude" respectively—two key features relevant to clinical and research applications. The extracted features are fed into classification models to distinguish between tremor and non-tremor videos. The Support Vector Machine (SVM), LSTM architecture and CNN-LSTM architecture are candidate classification models.

To compare our method with a benchmark, we reviewed the literature for similar published methods that automatically detect hand tremor. There were remarkably few studies that used machine learning methods applied to two-dimensional video data. Pintea et al. [16] employed the same video dataset that we use in the present study, but their method was not considered a suitable comparator as they focused on estimating the frequency of hand tremor rather than detecting hand tremor directly. In the same way, Uhrikova et al. [17, 18] estimated tremor frequency from video recordings, rather than classifying the presence of tremor. Other groups, such as Pang et al. [19] and Krupicka et al. [20] devised hand tremor detection methods but they used data from 3D videos (with depth information) [19] or 3D capture systems with two cameras and wearable reflective markers [20] to do so. Therefore, Soran et al.'s method [9] was selected as the most appropriate benchmark for the present study as they deployed machine learning based methods to classify tremor versus non-tremor and a 2D video dataset.

This project has two novel contributions. It is the first study to detect tremor using this unique combination methods of, Mediapipe to extract keypoints from hand region, an amplitude and frequency feature extraction, plus a neural network based classification. Secondly, it is the first study to detect and accurately classify tremor from videos with a cluttered background without any requirement for manual labelling of keypoints on the hand; all previous video-based tremor-detection studies necessitated a plain background to be used or require human labelling to identify the hands. The objectives of the study are to evaluate the accuracy of this new combination method to detect hand tremor in videos with a cluttered background and to clarify which sets of features, and which combinations of classification models provide the most accurate classification.

Previous research to automatically detect tremor using videos

Currently, clinicians detect tremor through observations, which is subjective. Researchers have begun to explore using new objective methods to automatically detect, and measure, tremor more accurately. In general, three methods have been used to detect and quantify hand tremor: using an accelerometer embedded in smartphones [21–23]; using wearable sensors attached to the hands [4–6] and using video recordings [9, 10]. Using video has several advantages over the other methods including being cheaper, non-touch, and requires less time and professional input.

For video-based hand tremor detection algorithms, Soran et al. [9] proposed a machine learning based computational method. They aimed to develop a hand tremor detection method which could automatically detect hand tremor from videos recorded on a plain background. Hand tremor was subtle, and therefore, it was a challenge to develop a method to distinguish between stable and tremulous hands. The proposed method involved three

steps including hand detection, feature extraction and classification. Personalized skin detection was the first step, then, optical flow of the pixels was used to extract directional motion features. The frequency of direction change was calculated by the Discrete Cosine Transform (DCT). Finally, Support Vector Machine (SVM) was used as the classification algorithm to classify hand tremor. For evaluation of the method, leave-one-out cross-validation (LOOCV) testing was performed with an accuracy of 95.4%. However, the method was limited by requiring a plain background which is not feasible in most applications. Roy, Rao and Anoumcia [24] used a similar system to detect hand tremor from videos. The approach was based on a machine learning algorithm. At the first step, input videos were divided into frames then Speeded-Up Robust Features (SURF) was applied to detect interest points in an image that were invariant to transformation. The Horn-Schunk optical flow algorithm and joint entropy were then applied to frames to extract features. A SVM algorithm was used to train a model to classify videos into tremor and non-tremor categories and the assumption made in their prediction system was that there could not be a visible movement of hands among five consecutive frames. Confusion matrix and precision-recall graphs represented the results of the experiments. The proposed algorithm was similar to Soran et al. [9]'s approach with different dataset used and they obtained 75.2% classification accuracy. In addition, Yohanandan et al. [10] used a video-based tremor assessment system, employing Blender v2.71 to track hand movements. This necessitated placing a marker on specific hand regions in the video to allow the tracking of hands. The frequency of hand tremor was calculated using the Fourier Transform and they obtained root-mean-square (RMS) of 0.3475 and 0.4373 for postural and kinetic tremor respectively. Pintea et al. [16] focused on calculating the frequency of hand tremor from videos when subjects were performing various tasks and found two approaches to estimate tremor frequency. In the first approach, hands were detected in each frame of the video, and frequency was estimated using a Lagrangian approach. The second approach was an Eulerian approach: the hands were first located, the large motion along the hand movement trajectory was removed and then video information overtime was encoded into phase information to estimate the tremor frequency.

The existing methods were all limited by the fact they required tightly regulated research settings with a plain background or manual labelling of keypoints on the hand. There remains a need for a video-based system that can detect tremor with a cluttered background without any requirement for manual labelling of keypoints on the hand. This would allow tremor quantification in more flexible environments such as clinics and people's own homes. Neural network based approaches have showed promising results in related fields such as gesture recognition. The applicability of neural network based approaches for detecting hand tremor in videos with a cluttered background is examined.

Methodology

The hand tremor detection algorithms contain three steps; see Fig. 1. The first step involves detecting the position of the hand. After the hands are detected, feature extraction algorithms are applied to extract key features from the hand region. The extracted features are recorded and applied to classifiers to discriminate between tremor and non-tremor. The advanced neural network architectures are used to perform the classification. The detailed explanation of the three steps is provided in this section.

Hand region detection

This section provides details on the implementation of hand region detection using a machine learning based model. The output of the hand region detection model is used to extract features and then fed into classification models.

As conventional hand detection algorithms involve processes that extract hand position data using color histograms, they are sensitive to change in lighting conditions or cluttered backgrounds. Therefore, we used a machine learning algorithm to train a model that could detect hands against cluttered backgrounds. MediaPipe [14] employs two models in a machine learning pipeline: a hand palm detection model and a hand landmark model. The hand palm detection model takes each frame as the input and outputs the boundary box containing the palm. The boundary box is treated as the input for the hand landmark model which returns the 21 keypoints

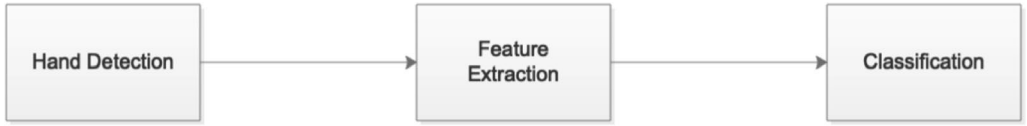


Fig. 1 Three steps included in the system of hand tremor detection

(see Fig. 2) of the hand as the output. The machine learning models involved in MediaPipe allow it to detect hands in a cluttered background and where hand or finger occlusions occur. Besides, the cropped hand region is processed by the hand palm model and fed into the hand landmark model, which dramatically minimizes the requirement of data augmentation including rotations, translation and scaling. In addition, in order to optimize the computational cost, the hand landmark model could provide the hand palm region based on the landmarks detected in the previous frame. The hand palm model is only applied to the frame when the hand landmark model could not detect the presence of the hand, which lowers the chance of losing track of hands and improves the hand detection accuracy.

Feature extraction

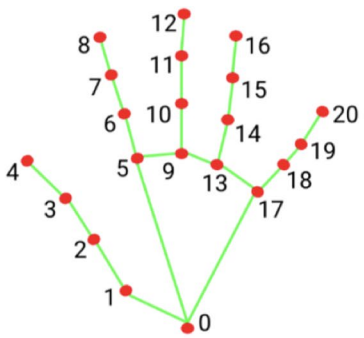
After the hand regions are detected and 21 landmarks are extracted from the input dataset, frequency of motion direction changes (MDC) and change in distance of hand movement within a fixed frame window are two sets of features extracted from the videos.

Frequency of motion direction changes (MDC)

The frequency of MDC is a critical feature of tremor [9]. Directional features are sensitive to rotation while directional changes features are rotation and translation invariant. Thus, the MDC is calculated for the purpose of tremor detection as it is less effected by rotation and translation of movements. To obtain the MDC of hands, the positions of 21 landmarks of hands from each frame of the videos are manipulated based on the geometry. Equation 1 shows the process of geometric manipulation from the x,y coordinates of each landmark to the angle, where symbol θ is the angle between the positive x direction and the line that joins the origin point and keypoint's coordinates.

$$\theta = \arctan\left(\frac{y}{x}\right) \tag{1}$$

The MDC is calculated within a fixed window containing five frames. Within the fixed window, the MDC of hands between the fifth and the first frames, between the fifth and second frames, between the fifth and the third frames and between the fifth and the fourth frames are calculated. Every five consecutive frames of the videos are analysed up to the last frame. The frequency of the MDC is the target feature to be collected from the system as a subtle movement is required to be detected. Thus, the next step converts the MDC into the frequency domain by using Discrete Cosine Transform (DCT) using Eq. 2 [26]. The output from the DCT is the frequency of



0. Wrist
1. Thumb carpometacarpal joint
2. Thumb metacarpophalangeal joint
3. Thumb interphalangeal joint
4. Thumb tip
5. Index finger metacarpophalangeal joint
6. Index finger proximal interphalangeal joint
7. Index finger distal interphalangeal joint
8. Index finger tip
9. Middle finger metacarpophalangeal joint
10. Middle finger proximal interphalangeal joint
11. Middle finger distal interphalangeal joint
12. Middle finger tip
13. Ring finger metacarpophalangeal joint
14. Ring finger proximal interphalangeal joint
15. Ring finger distal interphalangeal joint
16. Ring finger tip
17. Little finger metacarpophalangeal joint
18. Little finger proximal interphalangeal joint
19. Little finger distal interphalangeal joint
20. Little finger tip

Fig. 2 21 landmarks on left hand produced automatically by Mediapipe (Figure adapted from [25])

the MDC, which is one set of features extracted from the videos.

$$y(k) = 2\sum_{n=0}^{N-1} x(n)\cos\left(\frac{\pi(k)(2n+1)}{2N}\right) \quad (2)$$

In Eq. 2, symbols N and k are parameters involved in the Discrete Cosine Transform. The symbol N represents the length of the transform, which is 42 in this project. k = 0,1,...N-1 in the DCT-II formula and N is 42.

Change in distance of hand movement

Tremor amplitude is an important clinical feature to quantify and therefore the second tremor feature to be extracted is the change in distance of 21 landmarks of hands in a fixed window of five frames. The distance refers to the distance between the original point and the landmark position. The distances from the original point and 21 landmarks are calculated using Eq. 3 where x and y represent x coordinate of a keypoint and y coordinate of the keypoint respectively.

$$distance = \sqrt{x^2 + y^2} \quad (3)$$

To distinguish hand tremor from slow voluntary movements, the change in distance within a fixed window of five frames is calculated. Within the fixed window, the change in distance on each landmark of hands between the fifth and the first frames, between the fifth and the second frames, between the fifth and the third frames and between the fifth and the fourth frames are calculated. Every five consecutive frames of videos is analysed until the last frame of the video.

Classification models

The last step within the hand tremor detection system is classification. In this proof-of-concept study, three classification models are investigated. Different combinations of extracted features and classification models are also examined.

Support vector machine

The SVM model is investigated in our project [9]. The trained SVM model uses supervised data as input. The Gaussian Radial Basis Function (RBF) kernel is used for mapping features to higher-dimensional space. The RBF kernel has fewer parameters compared to the polynomial kernel, and therefore, it has a high accuracy of mapping especially for non-linear relationships. Then, the SVM classifier classifies the features. The trained model is used to predict whether the subject has hand tremor or not based on the training dataset.

CNN-LSTM and LSTM

CNN-LSTM architecture is the second classification model investigated. CNN-LSTM combines Convolutional Neural Network layers and Long Short-Term Memory (LSTM) architecture. For our project, we investigate a CNN-LSTM architecture with two Convolutional Neural Network layers and one Long Short-term Memory layer, which is modified based on Brownlee [27]'s work to detect hand gesture (Fig. 3). There are 64 neurons, 64 neurons and 100 neurons for the first convolution layer, the second convolution layer and the LSTM layer respectively.

The two Time Distributed Convolutional layers allow application of the same model to each of the six subsequences divided from the entire spatio-temporal

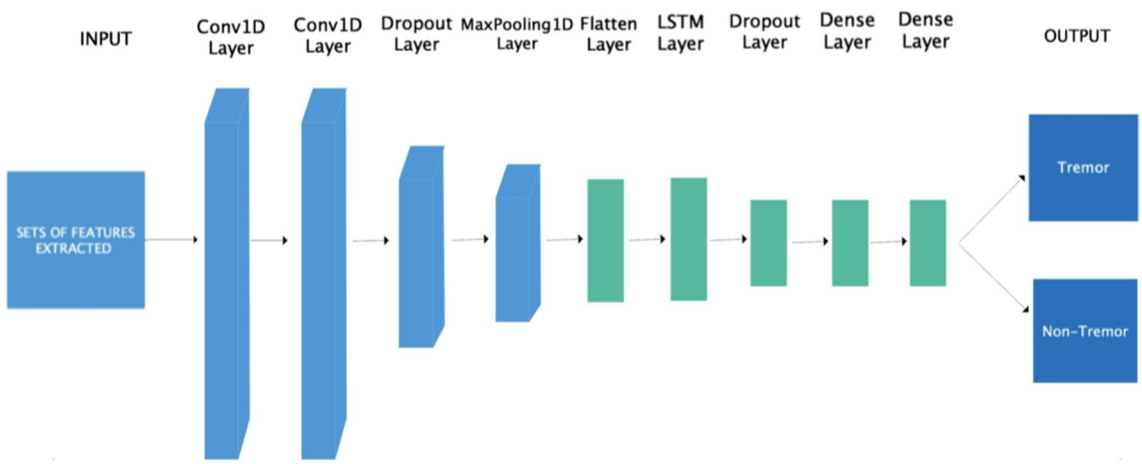


Fig. 3 CNN-LSTM Architecture for hand tremor detection. Conv1D layer stands for one-dimension Convolution Neural Network layer, LSTM layer stands for Long Short-Term Memory layer and MaxPooling1D layer stands for one-dimension max-pooling layer

input. Activation functions are used in convolutional layers to determine the effects of firing, or not firing, the neurons in the neural network model. The activation functions Sigmoid, Tanh and Rectified Linear Unit (ReLU) were investigated.

In addition to two convolutional layers and one LSTM layer included in the CNN-LSTM architecture, dropout layers, max-pooling layers and flatten layers are also involved in the architecture; see Fig. 3. The dropout layer randomly drops out nodes, which minimizes the effect caused by noise and reduces over-fitting. The proposed CNN-LSTM architecture contains two dropout layers: one is placed after the two convolutional layers and the other is placed after the LSTM layer. The dropout rate was set to be 0.5. In addition, the max-pooling layer reduces the resolution, or parameters, of the results given by the convolutional layer, thus reducing the computational load. By picking up the max value pixels using the filter in a max-pooling layer, we reduce over-fitting as the most active, or the most representative, pixels are selected to go to the next layer in the architecture. In the proposed CNN-LSTM architecture, the max-pooling layer is placed after the dropout layer following the two convolutional layers, with a pool size of 2. After the max-pooling layer of the CNN architecture, all extracted features go into the flatten layer so that extracted features are converted into the appropriate dimension to be fed into the LSTM layer.

A dense layer is placed after the last dropout layer to interpret the features extracted by the LSTM layer and feed them into the last prediction layer. The Softmax activation function is used in the prediction layer, outputting in the range of zero to one. In addition, the results of the activation function for each prediction categories are added to one. The output category with the highest value provides the tremor or non-tremor result; see Fig. 3.

The tremor classification results using solely LSTM, as an advanced neural network based classification model, are examined. LSTM architecture is capable of predicting spatio-temporal data input. Thus, the LSTM architecture is the candidate classification model in our hand tremor detection system. The LSTM architecture designed in our system contains three LSTM layers (Fig. 4). There are 64 neurons, 32 neurons and 32 neurons in the first, the second and the third LSTM layer respectively. Moreover, two dropout layers are added after the first and the second LSTM layer respectively. The dropout rate is set to be 0.2. The output of the model is tremor or non-tremor based on our research.

Experimental setup, results and discussion

Data description

We used the Technology in Motion Tremor Dataset (TIM-Tremor), an open-source dataset, collected in Leiden University Medical Centre, Netherlands [28]. The TIM tremor dataset comprises videos from 55 participants, 50 of whom have sufficient data for tremor quantification, and 5 of whom have no tremor recorded during the assessment.

The TIM-Tremor participants were asked to perform 21 actions including various poses and tasks classified as rest, postures, actions, distraction and entertainment; see Fig. 5.

The Microsoft KinectTM v2 sensor was used to record the participants. The KinectTM v2 sensor consists of one HD 1920×1080 RGB camera with a sampling rate of 30 frames per second. In addition, depth images were recorded by the sensor but we did not use depth images in our research. As we are aiming to detect hand

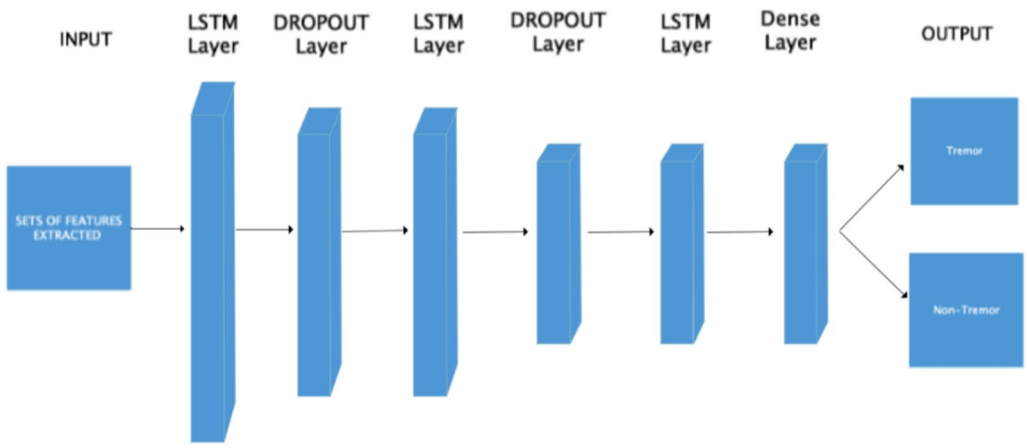


Fig. 4 LSTM Architecture for hand tremor detection

(a) Recorded tasks.

Task	Description
Rest	
Rest	Resting the arms on the chair handles.
Rest_in_supination	Resting the arms on the chair handles, hands in supination position.
Postures	
Hands_in_pronation	Both arms outstretched forward, hands in pronation position.
Thumbs_up	Both arms outstretched forward, thumbs up.
Top_top	Both hands in front of the chest with tips of the index fingers almost touching each other, elbows lifted sideways at approx. 90 degrees angle.
Weight	Affected arm outstretched forward, with a weight (1 kg) attached to the wrist.
Extra_pose	Holding a pose proposed by the medical expert to better visualize the tremor.
Actions	
Top_nose_left	Touching the top of the nose with the left index finger.
Top_nose_right	Touching the top of the nose with the right index finger.
Writing_left	Writing a given sentence with the left hand.
Writing_right	Writing a given sentence with the right hand.
Spiral_left	Drawing a spiral with the left hand.
Spiral_right	Drawing a spiral with the right hand.
Extra_writing	Extra writing task with a special pen, or diverging from the standard writing task with the affected hand.
Distraction	
Months_backward*	Naming the months backwards.
Counting*	Counting backwards (100 minus 7).
Finger_tapping*	Tapping with the index finger and thumb of the contralateral hand.
Playing_piano*	Moving the thumb of the contralateral hand across all fingers, from the index to the pinky finger and back.
Following*	Following a moving pointer with the index finger of the contralateral hand.
Entrainment	
2_Hz_higher*	Tapping with the contralateral hand in the rhythm of a flashing light, 2 Hz higher than the estimated tremor frequency of the affected hand.
2_Hz_lower*	Tapping with the contralateral hand in the rhythm of a flashing light, 2 Hz lower than the estimated tremor frequency of the affected hand.
*During these tasks, the affected hand was kept in the posture in which the tremor was most pronounced (i.e. arm on the chair handle, arm outstretched with hand in pronation or with thumb up, or hand in front of the chest).	

(b) Explanation.

Fig. 5 **a** 21 tasks are performed by subjects. **b** Short explanation for each task (Figure taken from [28])

tremor in videos with a cluttered background, the videos recorded by the RGB camera are selected to be the input of our proposed system. We defined a cluttered background as occurring when hands were not the only objects captured in the video frame. In the TIM-Tremor dataset, chairs, tables, computers, doors and other equipment in the lab were all recorded in the videos, which meant the videos of the hands had cluttered background.

As the hand region detection system involves a palm detection algorithm, the videos with the majority of hand palm presented were selected. As the hands were recorded relatively far away from the camera, the hand region detection algorithm could not detect the hand region with high accuracy. Thus, the hand regions with the cluttered background were cropped from the original videos to achieve higher accuracy in hand region detection. Each video in the dataset is approximately 30 s and, to enlarge the training dataset, these were trimmed into approximately 3-s videos.

For each patient's hand in the TIM-Tremor dataset, a labelling file is provided containing clinical ratings for tremor severity (ranging from 0 to 10) using the Bain and Findley Tremor Clinical Rating Scale. Tremor rating 0 indicates no tremor, 1–3: mild tremor, 4–6: moderate tremor, 7–9: severe tremor and 10: very severe tremor. We focused on distinguishing subjects with hand tremor (rating 1–10) and subjects without hand tremor (rating 0). Initially, 50 videos for each category (tremor vs non-tremor) were cropped and the size of the dataset to train the model was enlarged until we got a promising accuracy result. Finally, 189 cropped videos of hand tremor and 176 cropped videos without hand tremor were used as the dataset for our hand tremor detection system. The tremor ratings are treated as the ground truth. We compared the predicted results of the classification models with the ground truth to evaluate the model. The accuracy results from different combinations of sets of features and classification models were examined.

Evaluation method

The input dataset, of 189 cropped videos and 176 cropped videos for tremor and non-tremor category respectively, was divided into the training set and evaluation set. The hand region detection system, feature extraction methods and classification models were applied to the training set to train the hand tremor detection algorithms. The evaluation set was used to test the trained hand tremor detection algorithms.

Cross-validation settings were used to evaluate the system. Cross-validation is a model evaluation method to estimate the accuracy of the predictive model. The cross-validation helps to flag problems including over-fitting or selection bias. Furthermore, it tests on an independent subset which is not the training subset.

One round of cross-validation involves the process of partitioning a dataset into complementary subsets: one subset is used for training of the model, the other is used for evaluation of the model. To increase accuracy, multiple rounds of cross validation were utilized. The average result of the multiple rounds of cross-validation reduces the variability, and therefore, it increases the accuracy of the validation. In our proposed system, 10-fold cross-validation was applied to the different experimental scenarios.

Evaluation criteria

The accuracy, recall, precision and F1 score of 10-fold cross-validation were the evaluation criteria. The accuracy of the model is calculated based on Eq. 4.

$$Accuracy = \frac{TruePositive + TrueNegative}{TruePositive + FalsePositive + TrueNegative + FlaseNegative} \tag{4}$$

Recall is used to represent the proportion of correctly predicted positive results among all the actual positive results. Recall is calculated using Eq. 5.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{5}$$

Precision represents the proportion of correctly predicted positive results among all the predicted positive results, which is calculated using Eq. 6.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{6}$$

F1 score is a harmonic mean of recall and precision; it aims to balance the recall and precision of a model and is calculated using Eq. 7.

$$F1score = 2 * \frac{Precision * \mathrm{Re}call}{Precision + Recall} \tag{7}$$

The accuracy, recall, precision and F1 score, as evaluation metrics, are used to evaluate comparisons discussed in the experimental scenario section.

Results and discussion

Comparisons between Neural Network based classification models with different settings

Comparison results based on the LSTM model are presented in Appendix A. Two sets of features involved in the system include change in distance of hand movement (DIST features) and frequency of motion directional changes (MDC features).

The epoch size is the first setting that we examined for the CNN-LSTM based model. The accuracy results of the classification model with the increased epoch sizes shown in Fig. 6.

The accuracy increases rapidly with epoch size when the epoch size is less than 25 but only slowly when the epoch size is greater than 25. The epoch size determines the number of cycles of the whole dataset going through the neural network architecture. As we input the dataset to the neural network architecture repeatedly, the accuracy of the prediction model improves. However, there is a trade-off between the accuracy and training time.

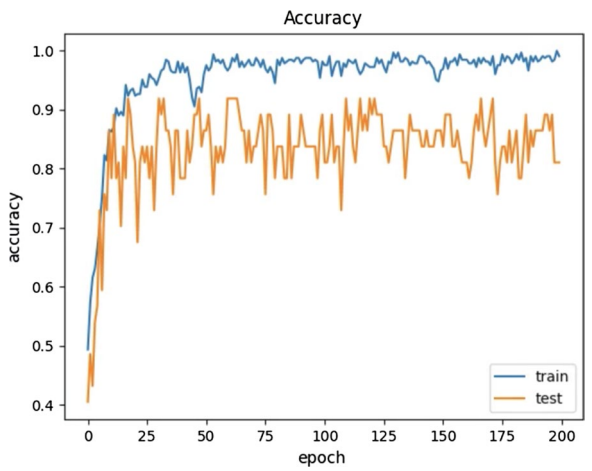


Fig. 6 Accuracy results of the CNN-LSTM model with increasing epoch sizes

The batch size is the second settings that we examined for the CNN-LSTM based model. The accuracy results for CNN-LSTM model with different batch sizes are shown in Fig. 7.

Based on the accuracy results of the CNN-LSTM model with different batch sizes, the accuracy results decrease with the increased batch sizes. The CNN-LSTM model with batch size of 8 has the highest average accuracy result. The batch size indicates the number of divisions of the dataset. Depending on the batch size, the dataset is divided into different parts and fed into the neural network model. The optimal batch size is determined with experiments.

The activation function was the third setting that we examined for the CNN-LSTM based model. The accuracy results of CNN-LSTM model with different activation functions are shown in Fig. 8.

According to the accuracy results of CNN-LSTM model with different activation functions, CNN-LSTM model with ReLU as an activation function has the highest average accuracy result. ReLU, comparing to other activation functions, does not fire all the neurons. ReLU fires the neuron only if the transformed value is greater than 0. Thus, it is widely used in the deep learning field and able to provide high prediction accuracy.

In conclusion, the CNN-LSTM model with epoch size of 200, batch size of 8 and ReLU as activation function provided the highest accuracy among all experiments. This setting of the CNN-LSTM model was used to perform further comparison between different sets of features and comparison between different classification models.

Comparisons between different sets of features combining each classification model

This section presents the results of comparisons between different sets of features combining SVM,

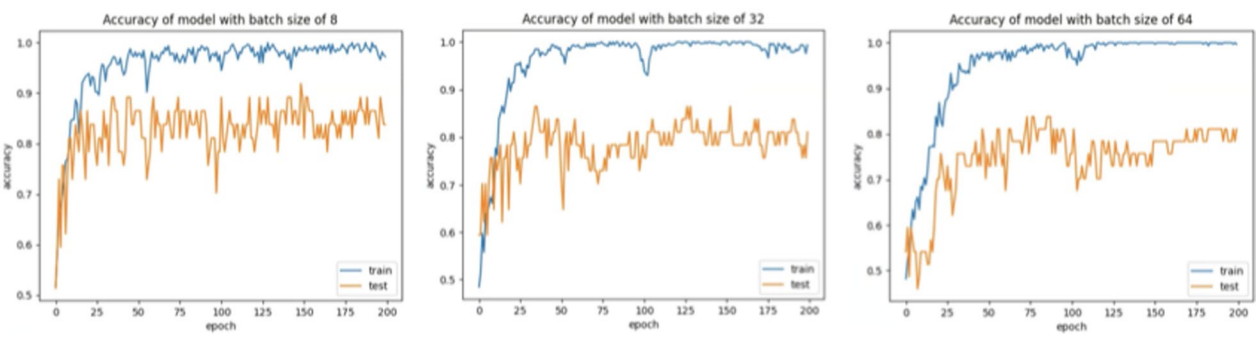


Fig. 7 Accuracy results of the CNN-LSTM model with different batch sizes

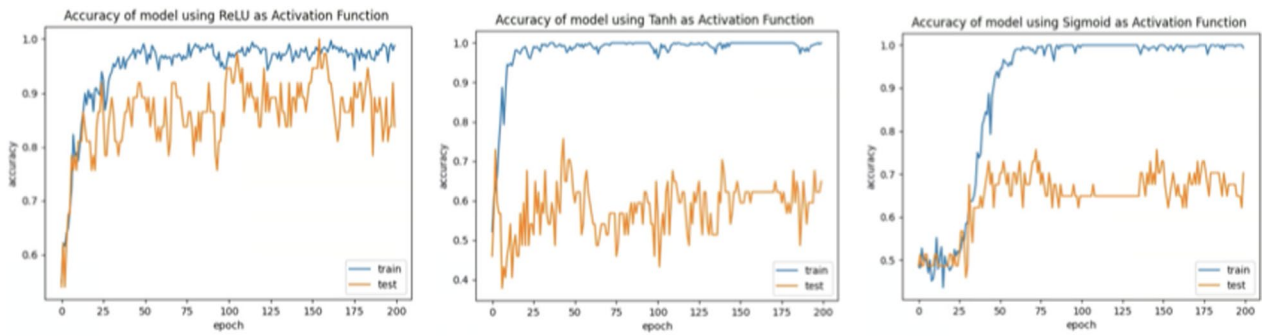


Fig. 8 Accuracy results of the CNN-LSTM model with different activation functions

Table 1 10-fold cross-validation results for configurations combining different sets of features with SVM model

Configurations combining different sets of features with SVM model	Average accuracy	Average precision	Average recall	Average F1 score
DIST with SVM model	0.53	0.51	0.63	0.55
MDC with SVM model	0.55	0.53	0.65	0.58
DIST + MDC with SVM model	0.56	0.53	0.67	0.59

Table 2 10-fold cross-validation results for configurations combining different sets of features with LSTM model

Configurations combining different sets of features with LSTM model	Average accuracy	Average precision	Average recall	Average F1 score
DIST with LSTM model	0.63	0.61	0.61	0.60
MDC with LSTM model	0.78	0.77	0.79	0.77
DIST + MDC with LSTM model	0.80	0.79	0.79	0.79

Table 3 10-fold cross-validation results for configurations combining different sets of features with CNN-LSTM model

Configurations combining different sets of features with CNN-LSTM model	Average accuracy	Average precision	Average recall	Average F1 score
DIST with CNN-LSTM model	0.72	0.72	0.67	0.68
MDC with CNN-LSTM model	0.79	0.79	0.78	0.78
DIST + MDC with CNN-LSTM model	0.81	0.81	0.79	0.80

LSTM and CNN-LSTM models respectively. The sets of features involved in the system include change in distance of hand movement (DIST features) and frequency of motion directional changes (MDC features).

The accuracy, precision, recall and F1 score of the 10-fold cross-validation results for the SVM model are presented in Table 1.

The accuracy, precision, recall and F1 score of the 10-fold cross-validation results for the LSTM model are presented in the Table 2.

The accuracy, precision, recall and F1 score of the 10-fold cross-validation results for the CNN-LSTM model are presented in the Table 3.

Based on the 10-fold cross-validation results of the configurations combining different sets of features with SVM model, the SVM model with two sets of features gives the highest average accuracy and F1 scores, of 0.56 and 0.59 respectively. According to the 10-fold cross-validation results of the configurations combining different sets of features with LSTM model, the LSTM model with

two sets of features gives the highest average accuracy and F1 scores, of 0.80 and 0.79 respectively. Based on the 10-fold cross-validation results of the configurations combining different sets of features with CNN-LSTM model, the CNN-LSTM model with two sets of features gives the highest average accuracy and F1 scores of 0.81 and 0.80 respectively.

As the change in distance of hand movement and frequency of hand directional changes are key features to distinguish between tremor and non-tremor, grouping two sets of features together to feed the classification model gives the highest prediction accuracy. Combining two sets of features provides a larger dataset for the classification models to learn from. Furthermore, the results show that when frequency of motion directional changes (MDC) was used as the sole feature there was a higher accuracy result for the classification of tremor than when change in distance of hand movement, or tremor amplitude (DIST), was used as the sole feature. One explanation for this finding is that the majority of videos comprised a low amplitude tremor that is more challenging to distinguish from non-tremor, than high amplitude tremor. In contrast, the frequency of hand tremor, captured by frequency of motion directional changes (MDC) feature, is not affected by amplitude of the tremor.

Comparisons between different classification models

This section presents the results of comparisons between different classification models. The sets of features involved in the system include change in distance of hand movement (DIST features) and frequency of motion directional changes (MDC features).

The accuracy, precision, recall and F1 score of 10-fold cross-validation of hand tremor detection algorithms combining different classification models are shown in Table 4. The SVM classification model, CNN-LSTM architecture and LSTM architecture were investigated in the hand tremor detection system. The hand tremor detection system proposed in Soran et al. [9]'s paper was chosen as the benchmark for our designed system as it combined the frequency of motion directional changes and a SVM classification model; furthermore, they also used similar machine learning based methods to detect and classify tremor. In Table 4, the average accuracy result of the SVM model with the frequency of motion directional changes as features is treated as a benchmark for our system.

The accuracy and F1 score are key indicators of the evaluation metrics. The advanced neural network based architectures including LSTM and CNN-LSTM architectures gave the highest accuracy and F1 score. The LSTM architecture, as an advanced neural network based architecture, has some advantages. First, it involves Forget Gate which determines whether to eliminate or keep the memory contributing to the results of prediction for each cell. In addition, each cell can perform a selection on which portion of the prediction would be the output of the cell. Thus, the models involving LSTM architecture would be expected to provide high prediction accuracy.

There is a difference between the accuracy of LSTM based approaches, including LSTM and CNN-LSTM architectures, with the benchmark approach. Comparing the LSTM based and CNN-LSTM based architectures, CNN-LSTM model had higher accuracy and F1 score.

Our study is the second study, to the best of our knowledge, to develop machine learning based models to detect the presence of tremor using two-dimensional videos (without depth information). We extracted frequency and amplitude features from the videos, and compared the accuracy results of different combinations of features with machine learning algorithms. Our approach extends the work in previous studies that solely extracted hand tremor frequency from video data [17, 18]. It also differs from previous studies that relied upon 3D video data (with depth information included) [19, 20] and thus potentially has greater reach clinically where non-depth cameras are more ubiquitous.

In summary, this is the first study that has shown tremor can be detected and accurately classified from videos with a cluttered background using a combination of Mediapipe, two extracted features of tremor (change in distance, or amplitude, of hand movement (DIST) and frequency of motion directional changes (MDC), and neural network based classification models. The designed system does not require manual labelling of keypoints on the hand and this is a major advantage for researchers and clinicians.

Table 4 10-fold cross-validation results for hand tremor detection algorithms with different classification models

Hand tremor detection algorithms with different classification models	Average accuracy	Average precision	Average recall	Average F1 score
MDC with SVM model (benchmark)	0.55	0.53	0.65	0.58
DIST + MDC with SVM model	0.56	0.53	0.67	0.59
DIST + MDC with LSTM model	0.80	0.79	0.79	0.79
DIST + MDC with CNN-LSTM model	0.81	0.81	0.79	0.80

It is important to acknowledge that a relatively small dataset has been used in this study. Thus, there are questions about the generalisability of this new combination method and it will be prudent to test this on independent larger dataset. This proof-of-concept study has shown the viability of combining these methods to automatically detect hand tremor but to develop it into a clinically useful application will require further research using clinical cohorts. Moreover, the feature extraction and classification steps are already integrated together but, for use in a clinical environment using videos as input and direct outputting the classification of tremor, further work will be required to integrate these with the keypoints extraction step too.

Conclusions

We found that hand tremor could be automatically detected despite a cluttered background with an accuracy of 80.6%. We used CNN-LSTM model with changes in the distance of hand movement and frequency of motion directional changes as features. We found that a CNN-LSTM model with epoch size of 200 and batch size of 8 gave the highest classification accuracy. ReLU, as a commonly used activation function in deep learning, was the activation function which gave the highest accuracy and F1 score according to the results of 10-fold cross-validation. For the comparisons between different sets of features, the configurations combining both changes in the distance of hand movement and frequency of motion directional changes as features with SVM, LSTM and CNN-LSTM models respectively gave the highest accuracy. Thus, two sets of features were critical features to distinguish between tremor and non-tremor. Lastly, for the comparison between accuracy results of different classification models, accuracy results showed a difference between the benchmark approach with the investigated neural network-based approaches (LSTM and CNN-LSTM).

The proposed approach detects hand tremor in videos with a cluttered background with high accuracy, and this was despite most videos comprising tremor with very small amplitude, which is far more challenging than using large amplitude severe tremors. This video-based tremor detection method has several potential applications and advantages over current capabilities. Users are not required to possess the professional knowledge to capture recordings of tremor automatically and the camera can be a standard 2D camera rather than specialist depth cameras. This opens up the possibility for clinicians and researchers to use their own smart phone cameras for tremor detection. In addition, so far, most tremor detection tools have been used in research settings with constraints such as a plain background [9, 19]. The current study shows that the technology can automatically detect tremor even when the background is cluttered. This is an exciting result as it raises the possibility that this method could be used in more "real world" settings such as home monitoring, healthcare clinics and more flexible research settings. Further studies, including with clinician data, will be required to assess this potential further. Moreover, the designed system has the potential to detect tremor while hands are moving as the neural network based architecture could learn from different training dataset and the trained model could perform tremor detection based on the condition presented in the training dataset.

Potential limitations are acknowledged; the approaches may require high computer processing power and memory to train the neural network model. Hidden layers are involved in the neural network model, which increases the complexity of the system. Moreover, the TIM-Tremor dataset contains data from 55 participants with only 5 of 55 participants labelled into the non-tremor category. The imbalanced number of participants with and without tremor may have affected the accuracy of the classifier model. In addition, the low-resolution videos recorded in the TIM-Tremor dataset affect the accuracy of the hand region detection system.

To further develop and validate this approach, it will be necessary to evaluate the system in independent larger cohorts that comprise a range of different tremors, for example Parkinson's, dystonic and Essential tremors. Before this system becomes a clinically-useful tool, it will also be important to assess whether variables such as ambient lighting, camera resolution and skin colour affects the results.

For future research, modifications to the method could allow real-time analysis. General features of the individual including age, gender and the presence of underlying diseases could be added as features to the classification model. Adding those features to the system could potentially increase the accuracy of the prediction of hand tremor. Modifications could be done to classification models to allow models to predict levels of tremor in addition to distinguishing between tremor and non-tremor.

Overall, this system shows promise at solving a common clinical and research problem but further studies are still required to translate it into the clinic.

Appendix A: Comparisons between LSTM models with different settings

See Tables 5, 6 and 7.

Table 5 10-fold cross-validation results LSTM model with different epoch sizes

LSTM model with different epoch sizes	Average accuracy	Average precision	Average recall	Average F1 score
DIST + MDC with LSTM model with epoch of 100	0.61	0.60	0.71	0.63
DIST + MDC with LSTM model with epoch of 150	0.63	0.62	0.71	0.64
DIST + MDC with LSTM model with epoch of 200	0.63	0.66	0.67	0.60

Table 6 10-fold cross-validation results LSTM model with different batch sizes

LSTM model with different batch sizes	Average accuracy	Average precision	Average recall	Average F1 score
DIST + MDC with LSTM model with batch size of 20	0.57	0.52	0.56	0.54
DIST + MDC with LSTM model with batch size of 18	0.60	0.59	0.69	0.62
DIST + MDC with LSTM model with batch size of 16	0.63	0.66	0.67	0.60

Table 7 10-fold cross-validation results LSTM model with different activation functions

LSTM model with different activation functions	Average accuracy	Average precision	Average recall	Average F1 score
DIST + MDC with LSTM model with Sigmoid activation function	0.74	0.73	0.75	0.73
DIST + MDC with LSTM model with Tanh activation function	0.80	0.79	0.79	0.79
DIST + MDC with LSTM model with ReLU	0.63	0.66	0.67	0.60

Author contributions
Conceptualization: XW, SG; Methodology: ST, XW, SG; Writing -original draft preparation: XW, SG, ST; Writing -review and editing: SG, JA, ST, QB; Supervision: SG, ST, QB, JA.

Funding
No funding was received to assist with the preparation of this manuscript.

Data availability
The dataset is publicly available.

Code availability
Code will be provided if requested.

Declarations

Conflict of interest
All the authors declared that they have no conflict of interest.

Author details
[1]Information and Communication Technology, School of Technology, Environments and Design, College of Sciences and Engineering, University of Tasmania, Hobort, TAS 7005, Australia. [2]Wicking Dementia Research and Education Centre, College of Health and Medicine, University of Tasmania, Hobort, TAS 7000, Australia.

Received: 6 February 2021 Accepted: 20 June 2021
Published online: 12 July 2021

References

1. Dorsey E, Sherer T, Okun MS, Bloem BR. The emerging evidence of the parkinson pandemic. J Parkinson's Dis. 2018;8(s1):S3–8.
2. EPDA: Motor symptoms tremor. 2021. https://www.epda.eu.com/about parkinsons/symptoms/motor-symptoms/tremor/.
3. Williams S, Fang H, Relton SD, Wong DC, Alam T, Alty JE. Accuracy of smartphone video for contactless measurement of hand tremor frequency. Mov Disord Clin Pract. 2021;8(1):69–75. https://doi.org/10.1002/mdc3.13119.
4. Cai G, Lin Z, Dai H, Xia X, Xiong Y, Horng SJ, Lueth TC. Quantitative assessment of parkinsonian tremor based on a linear acceleration extraction algorithm. Biomed Signal Process Control. 2018;42:53–62.
5. Rigas G, Gatsios D, Fotiadis D, Chondrogiorgi M, Tsironis C, Konitsiotis S, Gentile G, Marcante A, Antonini A. Tremor updrs estimation in home environment. In: 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC); 2016. pp. 3642–3645.
6. Rigas G, Tzallas A, Tsipouras M, Bougia P, Tripoliti E, Baga D, Fotiadis D, Tsouli S, Konitsiotis S. Assessment of tremor activity in the parkinson's disease using a set of wearable sensors. IEEE Trans Inf Technol Biomed. 2012;16(3):478–87.
7. Atas M. Hand tremor based biometric recognition using leap motion device. IEEE Access. 2017;5:23320–6.
8. Oktay A, Kocer A. Differential diagnosis of parkinson and essential tremor with convolutional lstm networks. Biomed Signal Process Control. 2020;56:101683.
9. Soran B, Hwang J, Lee S, Shapiro L. Tremor detection using motion filtering and svm. In: Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012); 2012. pp. 178–181.

10. Yohanandan S, Perera C, Jones M, Peppard RF, Perera T: Objective video-based tremor assessment for movement disorders using open-source software. In: 2017 IEEE Healthcare Innovations and Point of Care Technologies (HI-POCT), IEEE; 2017. pp. 192–195.
11. Hakim N, Shih T, Arachchi K, Priyanwada S, Aditya W, Chen Y, Lin C. Dynamic hand gesture recognition using 3dcnn and lstm with fsm context-aware model. Sensors. 2019;19(24):5429.
12. Tsironi E, Barros P, Weber C, Wermter S. An analysis of convolutional long short-term memory recurrent neural networks for gesture recognition. Neurocomputing. 2017;268:76–86.
13. Ullah A, Ahmad J, Muhammad K, Sajjad M, Baik SW. Action recognition in video sequences using deep bi-directional lstm with cnn features. IEEE Access. 2017;6:1155–66.
14. Google: Mediapipe(hands). GitHub repository. 2020. https://github.com/google/mediapipe
15. Kim A. Sign language recognition with rnn and mediapipe. GitHub repository. 2019. https://github.com/rabBit64/Sign-language-recognition-with-RNN-and-Mediapipe
16. Pintea L, Zheng J, Li X, Bank PJ, van Hilten JJ, van Gemert JC. Hand-tremor frequency estimation in videos. In: Proceedings of the European Conference on Computer Vision (ECCV); 2018. pp. 0–0.
17. Uhríková Z, Šprdlík O, Hoskovcová M, Komárek A, Ulmanová O, Hlaváč V, Nugent CD, Růžička E. Validation of a new tool for automatic assessment of tremor frequency from video recordings. J Neurosci Methods. 2011;198(1):110–3.
18. Uhríková Z, Sprdlik O, Hlavac V, Ruzicka E. Action tremor analysis from ordinary video sequence. In: 2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, IEEE; 2009 Sep 3. pp. 6123–26.
19. Pang Y, Christenson J, Jiang F, Lei T, Rhoades R, Kern D, Thompson JA, Liu C. Automatic detection and quantification of hand movements toward development of an objective assessment of tremor and bradykinesia in Parkinson's disease. J Neurosci Methods. 2020;333:108576.
20. Krupicka R, Szabo Z, Viteckova S, Ruzicka E. Motion capture system for finger movement measurement in Parkinson disease. Radio Eng. 2014;23(2):659–64.
21. Bazgir O, Habibi SAH, Palma L, Pierleoni P, Nafees S. A classification system for assessment and home monitoring of tremor in patients with Parkinson's disease. J Med Signals Sens. 2018;8(2):65.
22. Fraiwan L, Khnouf R, Mashagbeh A. Parkinson's disease hand tremor detection system for mobile application. J Med Eng Technol. 2016;40(3):127–34. https://doi.org/10.3109/03091902.2016.1148792.
23. Kostikis N, Hristu-Varsakelis D, Arnaoutoglou M, Kotsavasiloglou C. Smart phone based evaluation of parkinsonian hand tremor: Quantitative measurements vs clinical assessment scores. In: 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2014. pp. 906–909.
24. Roy K, Rao G, Anouncia S. A learning based approach for tremor detection from videos. In: 2013 IEEE Conference on Open Systems (ICOS), pp. 71–76 (2013) In: Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), 2012. pp. 178–181.
25. Mediapipe hands. https://google.github.io/mediapipe/solutions/hands.html. Accessed 31 Jan 2021.
26. Britanak V, Yip P, Rao K. Discrete cosine and sine transforms: general properties, fast algorithms and integer approximations. Amsterdam: Elsevier; 2010.
27. Brownlee J. Lstms for human activity recognition time series classification. 2020. https://machinelearningmastery.com/how-to-develop-rnn-models-forhuman-activity-recognition-time-series-classification/.
28. Bank P, Zheng J, Pintea S, PW O. Technology in motion tremor dataset: Tim-tremor. 4tu. centre for research data. dataset. 2019. https://doi.org/10.4121/uuid:522d14ed-3019-4206b49e-a4e674b6440a.

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