Emerging Technologies in Healthcare and Medicine, Vol. 116, 2023, 74–84
https://doi.org/10.54941/ahfe1004359

Early Characterization of Stroke Using Video Analysis and Machine Learning

Hoor Jalo[1], Andrei Borg[1], Elsa Thoreström[1], Marcus Lorentzon[1], Nathalie Larsson[1], Oskar Tryggvasson[1], Viktor Johansson[1], Petra Redfors[2], Bengt Arne Sjöqvist[1], and Stefan Candefjord[1]

[1]Chalmers University of Technology, Gothenburg 41296, Sweden
[2]Sahlgrenska University Hospital, Gothenburg 41345, Sweden

ABSTRACT

Stroke is one of the leading causes of death and disability worldwide and requires an immediate attention as the longer the patient is left untreated, the more severe its outcomes are. Enhancing access to optimal treatment and reducing mortality rates require improving the accuracy of stroke characterization methods in prehospital settings. This study explores how video analysis and machine learning (ML) can be leveraged to identify stroke symptoms based on the National Institute of Health Stroke Scale (NIHSS), with the goal of facilitating the prehospital management of patients with suspected stroke. A total of 888 videos were captured from the research group members, who mimicked stroke symptoms including facial palsy, leg and arm paresis, ataxia and dysarthria, following the criteria of the NIHSS. Multiple algorithms, utilized in earlier studies, were examined to predict these symptoms, and their performance was assessed using accuracy, sensitivity and specificity. The best method for detecting facial palsy was found using Histogram of Oriented Gradients (HOG) features in conjunction with Adaptive Boosting (AdaBoost), achieving accuracy, sensitivity and specificity values of 97.8%, 98.0% and 97.0%, respectively. The identification of arm paresis reached 100% on all metrics using a combination of MediaPipe and Support Vector Machine (SVM). For leg paresis, all algorithms had poor detection rates. The outcome of ataxia for both limbs varied. Google Cloud Speech-to-Text was used to detect dysarthria and reached 100% on all evaluation metrics. These findings suggest that video analysis and ML have the potential to assist in early stroke diagnosis, but further research is needed to validate this.

Keywords: Stroke, Machine learning, Video analysis, NIHSS, Prehospital diagnosis, Algorithms

INTRODUCTION

Stroke is ranked as the second leading cause of mortality and disability worldwide (Sirsat et al., 2020). Incidence increased by 70% between 1990 and 2019, due to the aging population (Feigin et al., 2022). Hemorrhagic stroke is caused by bleeding, and it accounts for 15% of stroke cases (Lumley et al., 2020). Ischemic stroke accounts for the majority of stroke cases (85%); it is caused by a blood clot obstructing the blood flow (Lumley et al., 2020). Large vessel occlusion (LVO) accounts for 24%–46% of ischemic stroke, occurring when the clot is located in the proximal part of a major intracerebral artery (Rennert et al., 2019). Blood flow can be restored by mainly two different treatments: thrombolysis for smaller ischemic stroke and thrombectomy for LVO, which is performed at what is called comprehensive stroke centers (CSC) (Lumley et al., 2020).

Patients with suspected stroke are for the most part transported to the hospital by ambulance (Mohammad, 2008), but accurate characterization of stroke in prehospital settings is still challenging. This is due to various factors such as time pressure, lack of diagnostic technologies and heterogenous clinical presentations (Lumley et al., 2020). Characterization of stroke starts with clinical stroke scales, which assess the different symptoms of stroke and are implemented using handwritten or digital protocols. The National Institute of Health Stroke Scale (NIHSS) is a comprehensive scale for the assessment of stroke. NIHSS score is the sum of 15 individually assessed items with a combined maximum of 42 points that categorizes the severity of stroke (0 = no stroke, 1–4 = minor stroke, 5–15 = moderate stroke, 16–20 = moderate to severe stroke and 21–42 = severe stroke) (Hage, 2011). Since it is a time-consuming test and not practical to be used in prehospital settings, simplified clinical scales are used instead, such as the Face Arm Speech Test (FAST). It however has a low level of specificity but a moderate-to-good level of sensitivity (Lumley et al., 2020).

Diagnosis of stroke can generally not be made until brain imaging is performed at the hospital (Lumley et al., 2020). Based on the prehospital stroke assessment, including the stroke scale score, the patient is usually transported to the nearest hospital (Fassbender et al., 2020). If the patient is diagnosed with LVO and the nearest hospital is not a CSC, the patient needs to be further transported to a CSC to perform thrombectomy (Fassbender et al., 2020). "Time is brain" emphasizes the fact that stroke is a highly time-critical condition, and early treatments are needed to improve stroke outcomes and reduce mortality (Vidale and Agostoni, 2018). Time window may be up to nine hours from symptom onset for thrombolysis and within 24 hours for thrombectomy (Jahan et al., 2019). The American Heart Association (AHA) guidelines called for improved prehospital stroke characterization tools and for research, including bypass algorithms, for triaging stroke patients to the most appropriate centers (Nicholls et al., 2022).

In recent years, machine learning (ML) and video analysis techniques have emerged as potential tools in healthcare and in the field of stroke assessment. A deep learning model was developed to analyze skeletal data from neurological examination videos to screen for signs of stroke (Jahan et al., 2019). Another study employed a multimodal deep learning approach to enhance the speed and accuracy of stroke diagnosis (Yu et al., 2020). These novel methods emphasize the potential of video analysis and ML for rapid and accurate stroke screening, with the potential to transform clinical practice.

The aim of this paper is to investigate the potential of ML and video analysis for early characterization of stroke by digitalizing parts of NIHSS items: facial palsy, upper extremities paresis, lower extremities paresis, limb ataxia (functional impairment as finger-nose ataxia and heel-knee ataxia) and dysarthria (speech disorder).

MATERIALS AND METHODS

Dataset

The target of this study is patients with potential stroke. Since no video data were available from patients with suspected stroke, video and audio data from healthy individuals mimicking stroke symptoms were collected. The dataset for this project was acquired by six research persons (authors AB, ET, ML, NL, OT and VJ). Research persons recorded themselves while mimicking stroke symptoms based on the NIHSS protocol. The videos were then reviewed for stroke symptom realism and approved by two senior stroke specialists involved in the project.

The videos were recorded in the sitting position using a Sony Alpha A6100 camera fixed on a tripod. A Sigma 56mm f/1.4 lens was used for the facial palsy recordings and Sigma 30mm f/1.4 lens for the other parts of the test. To improve the quality of the videos, a photography lighting kit (NEEWER 2 pack bi color 660 LED video light) was used.

Each symptom was recorded five times per person, and five times per body side for bilateral tests. The videos were recorded for different stroke symptoms and with varying severity from healthy to severe. The videos were cut down to one symptom imitation per clip using the DaVinci Resolve video editing program. The dataset consisted of 888 recordings (156 for facial palsy, 246 for arm paresis, 247 for leg paresis, 119 for finger-nose ataxia and 120 for heel-knee ataxia).

For dysarthria symptom, audio data was recorded for three severity levels while pronouncing certain words from NIHSS, and five recordings per each severity level were performed. The audio data were recorded using mobile phones (OnePlus Nord 2, Samsung Galaxy S22), microphones (Pre-Sonus M7) and computers (MacBook Pro 2015 and Windows Surface Book 2). The dataset consisted of 90 audio files to be used for training and evaluating the models.

Proposed Method

In this study, we propose a ML- and video analysis-based stroke prediction method. This section explains the data preprocessing methods, landmark extraction and the proposed classification models. The overall workflow of the proposed method consists of four steps: collecting videos, processing the videos using human landmark detectors, training ML classification models and evaluating the models' performance. The classification of stroke was performed as a binary classification problem, i.e., stroke vs. non-stroke. Video and audio files with stroke symptoms were labeled as 1, and without symptoms were labeled as 0. The data was consistently split as 75% training and 25% testing sets, unless specific deviations from this standard were stated. The performance of the classifiers was evaluated by the metrics of accuracy, sensitivity and specificity.

Facial Palsy

A two-step detection classifier was performed. The first step was to extract the facial landmarks and the second was to construct the ML classifiers using the extracted features. Histogram of Oriented Gradients (HOG) and Eigenface algorithms were used for face detection, which have been used in the detection of facial weakness and head pose (Aldridge et al., 2022; Hammadi et al., 2022). HOG extracts the features in an image that contain the most meaningful information by focusing on the shape and outline of the object (Aldridge et al., 2022). Eigenface is based on comparing differences between an image and the average values of all images in the dataset (Hammadi et al., 2022). All images must therefore have the same pixel size. Both algorithms are sensitive to unnecessary features of the images, such as background, size and color, and therefore pre-processing of images was required before they could be classified. MediaPipe (Bazarevsky et al., 2019) was first employed to create a "bounding box" highlighting the facial region. The image was then cropped to encompass only the area within the bounding box, eliminating any unneeded background information (Figure 1). The size of MediaPipe's bounding box around the face could vary. To derive uniform data for classification, the size of the images was converted to 200x250 pixels for HOG and 380x380 pixels for Eigenface. The images were also converted to grayscale.

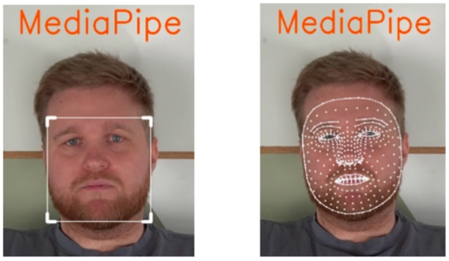

Figure 1: MediaPipe was used to crop facial region and remove unnecessary information.

In the HOG algorithm, each image is first divided into a grid of small, square cells consisting of a few pixels. The gradient is calculated for each pixel in the cell, whereupon a normalized histogram of all gradients within the cell is created. HOG features can then be extracted and used for the classification of the image (Figure 2). The gradients in the HOG image were then used as features, and a binary classification was performed where both eye and mouth symptoms were classified simultaneously. 3000 frames from the recorded videos with facial palsy were used in training and testing the algorithms. Adaptive Boosting (AdaBoost), Convolutional Neural Network (CNN), Deep Neural Network (DNN) and Support Vector Machine (SVM) were trained and evaluated. Different combinations of layers and neurons were tested with DNN to obtain the maximum possible accuracy. A network with three hidden layers was used to obtain the final results with 64, 128 and 64 neurons in each layer. Two convolutional layers were used in CNN. 20 epochs were used in the training of neural networks and a learning rate of 0.001. When classifying with SVM, the following parameters were used: gamma = 0.5 and C = 0.1.

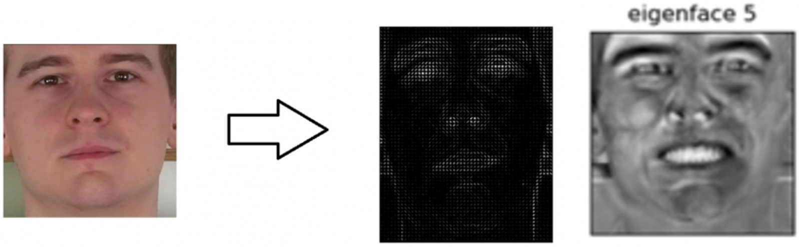

Figure 2: Facial landmarks extraction with HOG and eigenface algorithms.

In Eigenface, Principal Component Analysis (PCA) was applied to the preprocessed facial images with 10 components as it captured good variations (Figure 2). CNN was not used for Eigenface image classification since Eigenface breaks down the image into a few important components, thus the CNN does not get a relevant amount of data to work with. DNN, AdaBoost and SVC were however trained and evaluated with the same settings as for the HOG algorithm. 20 epochs were used in the training of neural networks and a learning rate of 0.001.

Recurrence plots (RP) (Lee et al., 2022) is a method of visualizing recurring patterns and dynamics in a time series, making it possible to determine the degree of entropy in a non-linear or complex system. An important parameter when producing RP is the threshold value, which should be chosen carefully since a significantly large threshold leads to important details being lost in the pattern and significantly small threshold makes the PR sensitive to noise (Lee et al., 2022). A threshold of 0.2 was used after a number of careful tests.

RP were tested on facial palsy on the upper and lower half of the face. The facial points for the upper and lower eyelids and for the corners of the mouth and the ears were detected using MediaPipe as it is capable of tracking 468 facial points (Figure 1), providing detailed information about the shape and movements of a person's face (Bazarevsky et al., 2019). The distance between the upper and lower eyelids of each eye generated a recurrence plot, which was then combined into a common image with a size of 930x465 pixels. In the same way, the distance between the corner of the mouth and nearby ear generated two recurrence plots which were then linked together into a common image. The combined image was then analyzed by the binary classifiers DNN, CNN and Residual Neural Network (ResNet). For ResNet (Figure 3), different sizes of hidden layers were tested: ResNet-50, ResNet-101 and ResNet-152.

RP read data from an entire time series, which means the entire video, only 30 plots for normal expressions and 60 plots for mimicked stroke symptoms were created from the collected videos. Due to the limited amount of data, augmentation was used to expand the amount of data and to avoid overfitting. For augmentation, the techniques of rotation, width shift, height shift, zoom, horizontal flip and shear were used, and augmentation increased the amount of training data to about 3500 images in total. RP were divided into 80% training, 15% validation and 5% test sets. Normalization was performed using the Keras library (Keras).

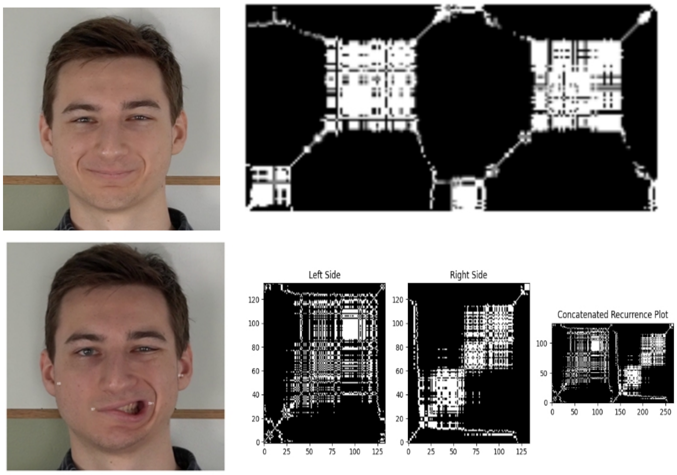

Figure 3: RP for a symmetrical smile with no stroke symptoms (top) and for weakness in the left corner of the mouth (bottom).

A grid search method was used to identify the best combinations of hyperparameters for the neural networks. Different numbers of layers, dense layers, epochs and learning speed were tested for CNN, DNN and ResNet. The choice of batch size, whether eight or 16, was determined based on the specific performance requirements of the model and the available graphics card memory. For DNN, two dense layers were used with ReLU as activation functions, where the last layer had a binary output of two neurons with a SoftMax function. This output layer was used in all three investigated neural network classifiers. The DNN classifier was trained in 15 epochs with a learning rate of 0.0001 and a categorical cross entropy loss function. The CNN model consisted of three convolutional layers, each with a max-pooling layer of 4x4 size and a stride value of 4. The CNN was trained in 10 epochs with a learning rate of 0.0001 and a categorical cross entropy loss function. For ResNet, 152 hidden layers and a parameter set similar to CNN were eventually used, and 5 epochs was set as higher epochs raise risks of overfitting.

Extremity Paresis and Limb Ataxia

Several popular approaches are used in multi-human pose estimation, and two of them have been used in this study. First, OpenPose (Cao et al., 2021), a real-time Python library that detects key points in the human body, face and feet using a CNN with confidence maps and part affinity fields. MediaPipe was also used since it has the ability to identify 33 key points in the human body (Bazarevsky et al., 2019). OpenPose and MediaPipe were evaluated for detecting body movements from videos of arm and leg paresis and ataxia. Both algorithms produce a key point for both the hand and the foot, as well as coordinates for where in the image they are located. For limb paresis, the difference in the y-axis between the limb and the top of the image was used to quantify the patient's motion. For each video, a list of limb positions was created based on the y-coordinate of the key point, which then showed how the patient had moved during the video (Figure 4). For ataxia, the Euclidean distance between the hand and the nose of the person in the image was calculated. The classification algorithms were then trained on how that distance changed between the images in a video.

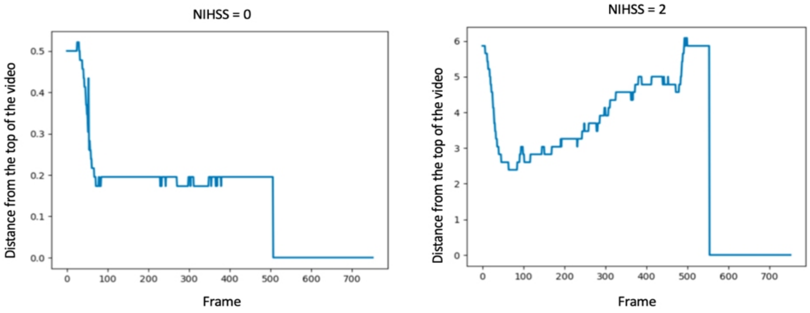

Figure 4: Two severity levels of arm paresis (NIHSS = 0 and 2, respectively).

DNN, SVM and ResNet-152 were trained and evaluated for limb paresis and ataxia. SVM parameters were gamma = 0.2 and C = 0.8. Various number of layers and learning rates were tested, and the best performing combination was two convolutional layers with a max-pooling layer. The DNN was made up of two dense layers with 64 and 32 neurons and a dense SoftMax layer with two neurons. DNN classifier was trained in seven epochs, while ResNet was trained in three.

Dysarthria

To test and evaluate this digitalization of this NIHSS item, a list of words was repeated, where they were pronounced correctly if no stroke symptoms were present and slurred and unrecognized speech for mimicking stroke. Symptoms of dysarthria were mimicked in three stages: normal speech, slurred speech and inability to formulate words. A binary classification was carried out where normal speech was separated from the speech difficulties.

The recorded audio files were then transcribed using Google Cloud Speech-to-Text using the Speech recognition service, which was previously used in the detection of speech inabilities in stroke patients (Yu et al., 2020). A string of the recordings was returned by the model, which was then translated into a Bag-of-Words (BOW) representation for the classification algorithms to interpret. Longer pauses in the recorded speech were also detected with the pydub library in Python. SVM, AdaBoost and DNN were used to classify the audio files.

RESULTS AND DISCUSSIONS

Facial Palsy

Two different methods were tested for the detection of facial palsy with a binary classification that obtained an accuracy of over 90% for both methods: detection in real-time and over a time series. Eigenface and HOG classified the videos in real-time as the algorithms go frame by frame in the video and determine if there are symptoms of stroke. RP instead looks at a span of time, in this case an entire video, where the entire clip was classified as stroke or non-stroke.

SVM and AdaBoost were difficult to apply for HOG and RP as they use whole images as input for classification. AdaBoost and SVM work optimally when classifying with relatively few features, and the complexity becomes high with images, which results in long training and classification time. In HOG classification, SVM took 4.5 times and AdaBoost 60 times the time by neural network classifications. Classification with AdaBoost and SVM was thus excluded for RP as training lasted too long. The algorithms however worked well for Eigenface, which uses only a few components to represent an image (Table 1).

The best results were obtained by HOG combined with AdaBoost (accuracy of 97.8%). For RP, better performance was obtained for all algorithms when testing for the lower part of the face, and the best performance was achieved by the CNN classifier with the highest accuracy, sensitivity and specificity (Table 1). Overall, real-time detection methods performed better in classification, where a sensitivity of 98% was obtained with HOG, which means that signs of stroke were misclassified in 2% of cases. Real-time detection had however more training data for each classification where every frame in a video is reviewed, while a time series only gets one plot per video. In the study, 3000 data points were used for Eigenface and HOG and 90 data points for the RP (3500 with augmentation). Despite this, the RP got good results with an accuracy, sensitivity and specificity of over 90% using CNN.

Extremity Paresis and Limb Ataxia

When training and evaluating OpenPose and MediaPipe on videos mimicking stroke symptoms (limb paresis and ataxia), both algorithms struggled to detect leg key points accurately, especially in mimicking leg paresis. The leg was lifted towards the camera, which resulted in the hip, knee and foot being at the same point in the video, which the algorithms failed to detect (Figure 5). The detection of the leg was successful in some images, but not others. However, MediaPipe performed better than OpenPose in the case of ataxia (Table 1). OpenPose achieved 56% accuracy for ataxia and 46.7% for leg paresis, while MediaPipe performed better with 90% accuracy for ataxia but only 46.7% for leg paresis. MediaPipe was faster compared to OpenPose, where a preprocessing of 10-second video lasted for 15 and 100 seconds, respectively.

For arm paresis, the best performance was obtained by MediaPipe together with SVM, where 100% accuracy, sensitivity and specificity were achieved. MediaPipe performed generally better than OpenPose with all the classification algorithms (Table 1). A specificity of 0% for both methods was obtained by ResNet, indicating that all data was classified as stroke. That may be because better preprocessing of data was required to create RP for the specific case of paresis and ResNet.

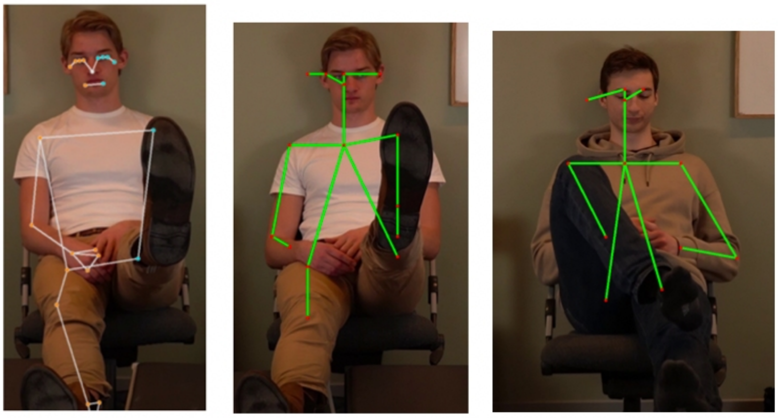

Figure 5: Leg is not detected by MediaPipe (left) and OpenPose for both leg paresis (middle) and ataxia (right).

For ataxia, SVM with all algorithms achieved 0% specificity, meaning that no non-stroke ataxia was correctly identified. MediaPipe with DNN was the best combination for both finger-nose and heel-knee ataxia. Due to the small amount of training and test data, further research would be required to verify this result. The result for arm ataxia differed from arm paresis, where the accuracy was overall lower for all classifiers. This could be because paresis is a linear movement while ataxia is a more complex movement, making it more difficult to distinguish.

Dysarthria

For dysarthria, all classifiers performed perfectly (Table 1) across all the evaluation metrics. Considering the nature of this classification problem, the algorithms are expected to perform perfectly since the features used indicate the amount of times a given word appeared in the recording. The algorithms however do not consider patients with natural speech disorders or heavy accents.

Strengths and Limitations

The main strength of this study is it investigates the digitalization of the NIHSS test with the use of ML and video analysis, which has the potential to provide faster diagnosis thus treatment. The study has however several limitations including that no clinical patient data were included, but the algorithms were rather trained and evaluated on data collected from healthy individuals. A limited amount of training data was used, and all participants shared similar characteristics like age, skin color, and hair color, which has the potential to introduce inherent bias. Furthermore, the study lacks external validation of its findings.

Table 1. Different classifiers evaluation (best classification is highlighted in bold). RP is presented as lower/upper half of the face.

NIHSS item	Algorithm	Classifier	Accuracy (%)	Sensitivity (%)	Specificity (%)
Facial palsy	HOG	AdaBoost	**97.8**	**98.0**	**97.0**
		CNN	94.3	97.0	91.0
		DNN	**97.8**	**98.0**	96.0
		SVM	97.6	**98.0**	95.0
	Eigenface	AdaBoost	96.6	97.0	95.0
		DNN	96.2	97.0	92.0
		SVM	96.0	97.0	91.0
	RP	CNN	94.1/88.2	96.2/88.9	92.0/87.5
		DNN	80.4/70.6	80.8/75.0	80.0/69.2
		ResNet	84.3/66.7	84.6/33.3	84.0/66.7
Arm paresis	OpenPose	DNN	88.1	95.7	61.5
		SVM	93.0	97.0	77.0
		ResNet	78.3	100	0
	MediaPipe	DNN	91.0	97.7	22.2
		SVM	**100**	**100**	**100**
		ResNet	82.7	**100**	0
Ataxia (finger-nose)	OpenPose	DNN	76.7	64.3	87.5
		SVM	47.0	**100**	0
		ResNet	53.3	0	**100**
	MediaPipe	DNN	**86.6**	71.4	93.8
		SVM	47.0	**100**	0
		ResNet	60.0	0	**100**
Ataxia (heel-knee)	MediaPipe	DNN	**90.0**	85.7	93.8
		SVM	47.0	**100**	**100**
		ResNet	70.0	14.3	0
DYSARTHRIA	Google Speech-to-Text	DNN	**100**	**100**	**100**
		SVM	**100**	**100**	**100**
		ResNet	**100**	**100**	**100**

CONCLUSION

All algorithms succeeded at detecting facial paresis in the videos, both in real-time and for time series. Best accuracy was achieved with HOG features in combination with the AdaBoost classifier. There were indications of recurrence plots being well-suited for the detection of facial paresis, but further data are required to confirm this.

When it comes to the detection of body movements, the results varied. For the detection of arm paresis, the combination of MediaPipe and SVM achieved the highest accuracy. Accuracy for detecting leg paresis was low, which could depend on the lighting and camera angle during video filming. Results for detecting ataxia varied substantially and need to be investigated in further studies.

All algorithms were effective at detecting imitated speech disorders. This indicates that they might detect dysarthria in real patients. However, the current project did not account for speech variations like natural speech errors and accents.

In conclusion, the findings suggest that digitizing parts of the NIHSS is possible, and that video analysis and ML have the potential in the early detection of various stroke symptoms. This however comes with several challenges and considerations, and further studies are thus required to validate and build upon these results.

ACKNOWLEDGMENT

The authors would like to acknowledge neurologist Annika Nordanstig at Sahlgrenska University Hospital for her valuable contribution to assessing and validating the videos depicting individuals mimicking stroke symptoms.

REFERENCES

Aldridge, M., McDonald, M., et al. (2022) Human vs. Machine Learning Based Detection of Facial Weakness Using Video Analysis; Front Neurol.

Bazarevsky, V., Kartynnik, Y., et al. (2019) BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs; arxiv.

Cao, Z., Hidalgo, G., et al. (2021) OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. IEEE Trans. Pattern Anal. Mach; Intell.

Fassbender, K., Walter, S., et al. (2020) Prehospital stroke management in the thrombectomy era; Lancet Neurol.

Feigin, V. L., Brainin, M., et al. (2022) World Stroke Organization (WSO): Global Stroke Fact Sheet 2022; Int J Stroke.

Hage, A. (2011) The NIH stroke scale: a window into neurological status; Nurs Spectr.

Hammadi, Y., Grondin, F., et al. (2022) Evaluation of Various State of the Art Head Pose Estimation Algorithms for Clinical Scenarios; Sensors (Basel).

Jahan, R., Saver, L. et al. (2019) Association Between Time to Treatment With Endovascular Reperfusion Therapy and Outcomes in Patients With Acute Ischemic Stroke Treated in Clinical Practice; JAMA.

Keras: Deep Learning for humans [WWW Document]. URL https://keras.io/

Lee, T., Jeon, T., et al. (2022) Deep-Learning-Based Stroke Screening Using Skeleton Data from Neurological Examination Videos; J Pers Med.

Lumley, H. A., Flynn, D., et al. (2020) A scoping review of pre-hospital technology to assist ambulance personnel with patient diagnosis or stratification during the emergency assessment of suspected stroke; BMC Emergency Medicine.

Mohammad, M. (2008) Mode of Arrival to the Emergency Department of Stroke Patients in the United States; J Vasc Interv Neurol.

Nicholls, K., Ince, J., et al. (2022) Emerging Detection Techniques for Large Vessel Occlusion Stroke: A Scoping Review; Front Neurol.

Rennert, C., Wali, R., et al. (2019) Epidemiology, Natural History, and Clinical Presentation of Large Vessel Ischemic Stroke; Neurosurgery.

Sirsat, S., Fermé, E., Câmara, J. (2020) Machine Learning for Brain Stroke: A Review; Journal of Stroke and Cerebrovascular Diseases.

Vidale, S., Agostoni, E. (2018) Prehospital stroke scales and large vessel occlusion: A systematic review; Acta Neurol Scand.

Yu, M., Cai, T., Huang, X., et al. (2020) Toward Rapid Stroke Diagnosis with Multimodal Deep Learning; Springer International Publishing.