

Article

Classification of Activities of Daily Living Based on Grasp Dynamics Obtained from a Leap Motion Controller

Hajar Sharif ^{*}, Ahmadreza Eslamina , Pramod Chembrammel [†] and Thenkurussi Kesavadas ^{*,†}

Department of Mechanical Science and Engineering, Health Care Engineering Systems Center, The Grainger College of Engineering, University of Illinois at Urbana-Champaign, Champaign, IL 61820, USA

^{*} Correspondence: hsharif2@illinois.edu (H.S.); tkesavadas@albany.edu (T.K.)

[†] These authors were affiliated with University of Illinois at Urbana-Champaign while study was performed.

Abstract: Stroke is one of the leading causes of mortality and disability worldwide. Several evaluation methods have been used to assess the effects of stroke on the performance of activities of daily living (ADL). However, these methods are qualitative. A first step toward developing a quantitative evaluation method is to classify different ADL tasks based on the hand grasp. In this paper, a dataset is presented that includes data collected by a leap motion controller on the hand grasps of healthy adults performing eight common ADL tasks. Then, a set of features with time and frequency domains is combined with two well-known classifiers, i.e., the support vector machine and convolutional neural network, to classify the tasks, and a classification accuracy of over 99% is achieved.

Keywords: Leap Motion Controller; activities of daily living; hand grasps classification



Citation: Sharif, H.; Eslamina, A.; Chembrammel, P.; Kesavadas, T. Classification of Activities of Daily Living Based on Grasp Dynamics Obtained from a Leap Motion Controller. *Sensors* **2022**, *22*, 8273. <https://doi.org/10.3390/s22218273>

Academic Editor: Sašo Blažič

Received: 25 September 2022

Accepted: 24 October 2022

Published: 28 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Many neurological conditions lead to motor impairment of the upper extremities, including muscle weakness, altered muscle tone, joint laxity, and impaired motor control [1,2]. As a result, common activities such as reaching, picking up objects, and holding onto them are compromised. Such patients will experience disabilities when performing activities of daily living (ADLs) such as eating, writing, performing housework, and so on [2].

Several evaluation methods are commonly being used to assess problems in performing ADLs [3–5]. Despite the wide application of these methods, all of them are subjective techniques, i.e., they are either questionnaires or qualitative scores assigned by a medical professional [3,5]. We hypothesize that providing a more quantitative metric could enhance the evaluation of the rehabilitation progress and lead to a more efficient rehabilitation regimen tailored to the specific needs of each individual patient.

For instance, a quantitative methodology could help to defer the plateau in the patient's recovery. 'Plateau' is a term that is used to explain a stage of stroke recovery at which functional improvement is not observed (see Figure 1) and is determined by clinical observations, empirical research, and patient reports. In spite of the importance of plateau time as an indication of the time to discharge a patient from post-stroke physiotherapy, researchers have questioned the reliability of current methods for determining the plateau [6,7]. Demain et al. [6] implemented a standard critical appraisal methodology and found that the definition of recovery is ambiguous. For instance, there is a 12.5–26 week variability in plateau time for ADLs. A few parameters have been attributed with causing such inconsistency, among which, the qualitative nature of the assessment metrics can be mentioned [6–8]. An early and unnecessary discharge from physiotherapy can leave the patient with a permanent, yet potentially preventable, disability and having a more reliable technique to indicate the start of the plateau could help to determine the time at which to adjust the rehabilitation regimen and minimize neuromuscular adaptations which, in turn, can delay the plateau [8].

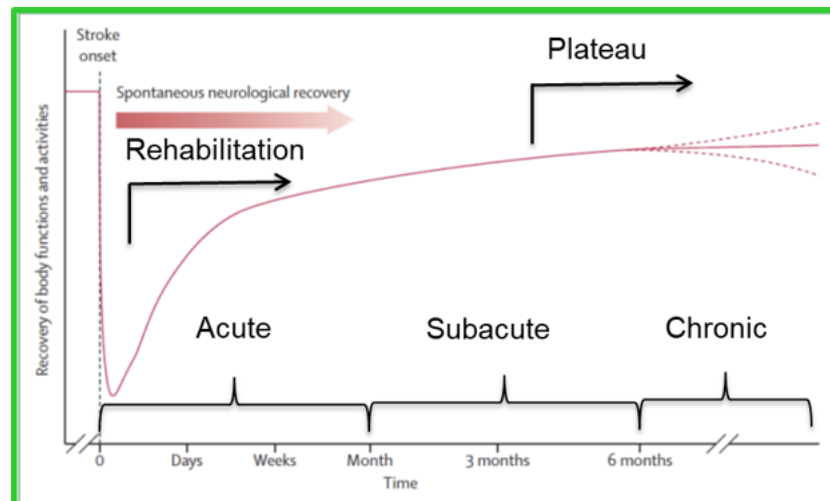


Figure 1. Stroke timeline [3].

The term “Activities of Daily Living” has been used in many fields, such as rehabilitation, occupational therapy, and gerontology, to describe a patient’s ability to perform daily tasks that allow them to maintain unassisted living [9]. Since this term is very qualitative, researchers have proposed many subcategories of ADL, such as physical self-maintenance, activities of daily living, and instrumental activities of daily living [10], to assist physicians or occupational therapists in evaluating the patient’s ability to perform ADLs in a more justifiable fashion [9,11,12].

A fundamental step towards developing a quantitative ADL assessment methodology is to distinguish different ADL tasks based on hand gesture data. Based on the hardware applied to detect hand gestures, hand gesture recognition (HGR) methods can be divided into sensor-based and vision-based categories [13]. In sensor-based methods, the equipment used for data collection is exposed to the user’s body, whereas in vision-based techniques, different types of cameras are used for data acquisition [14,15]. Vision-based methods do not interfere with the natural way of forming hand gestures; however, several factors such as the number and positioning of cameras, the hand visibility, and algorithms applied on the captured videos can affect the performance of these techniques [13].

The Leap Motion Controller (LMC) is a marker-free vision-based hand-tracking sensor that has been shown to be a promising tool for HGR applications [16,17]. Several researchers have used the LMC to detect signs using hand gestures for American [18,19], Arabic [20–23], Indian [24–26], and other sign languages [27–32]. LMC has applications in education [33] and navigating robotic arms [34,35]. Researchers have investigated LMC applications in medical fields [36,37] including, but not limited to, upper extremity rehabilitation [38–41], wheelchair maneuvering [42], and surgery [43,44]. Bachmann et al. [45] reviewed the application of LMC for a 3D human–computer interface, and some studies have focused on the use of LMC for real-time HGR [46,47].

In this study, we used data collected from healthy subjects to develop the first stage of quantitative techniques that have a wide range of applications in improving the outcomes of assessments of many common neurological conditions. We demonstrated two classification schemes based on SVM and CNN that can efficiently classify ADL tasks. These classifiers use the features extracted by existing feature engineering methods from the collected data. In addition, we generated a dataset containing hand motion data collected using LMC while the participants performed a variety of common ADL tasks. We tested the performance of the proposed classification schemes using this dataset.

The tasks selected from this dataset included a variety of ADLs associated with physical self-maintenance, e.g., utilizing a spoon, fork, and knife, and activities of daily living, e.g., writing. In addition, based on Cutkosky grasp taxonomy, the tasks in this study include precision grasps, such as holding a pen, spoon, and spherical doorknob as well as

power grasps like holding glass, a knife, and nail clippers [5,48,49]. These tasks involve diverse palm/finger involvement and facilitate the analysis of hand grasp over the entire range of motion that is typically used in ADLs.

2. Materials and Methods

2.1. Subjects and Data Acquisition

In this study, an LMC was employed to collect data from the dominant arm of the participants while they performed tasks. The LMC is a low-cost, marker-free, visual-based sensor that works based on the time-of-flight (TOF) concept for hand motion tracking. It contains a pair of stereo infrared cameras and three infrared LEDs. Using the infrared light data, the device creates a grayscale stereo image of the hands. As shown in Figure 2, the LMC is designed to either be placed on a surface, e.g., on an office desk, facing upward or be mounted on a virtual reality headset. To collect the ADL data, a 7-degrees-of-freedom robotic arm, i.e., Cyton Gamma 300 [50], was used to hold the LMC at an optimum position to minimize occlusion. The experimental setup and hand model in the LMC with the global coordinate system (GCS) are provided in Figure 3a,b, respectively. The LMC reads the sensor data and performs any necessary resolution adjustments in its local memory. Then, it streams the data to Ultraleap's hand tracking software on the computer via a USB. It is compatible with both USB 2.0 and USB 3.0 connections. LMC's interaction zone is between 10 cm and 80 cm from the device and has a $140^\circ \times 120^\circ$ typical field of view, as shown in Figure 4 [32,51,52].

Nine healthy adults with intact hands, including three females and six males, were recruited to participate in this study, and informed consent was obtained from all participants. The age range of the participants was 25–62 years with an average of 37 years. This study was approved by the Institutional Review Board office of University of Illinois at Urbana-Champaign, and there were no limitations in terms of occupation, gender, or ethnicity when recruiting the participants.

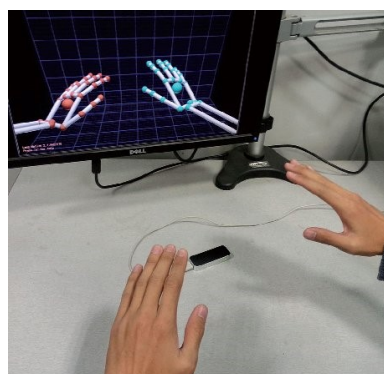
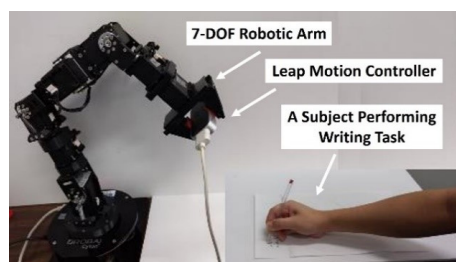
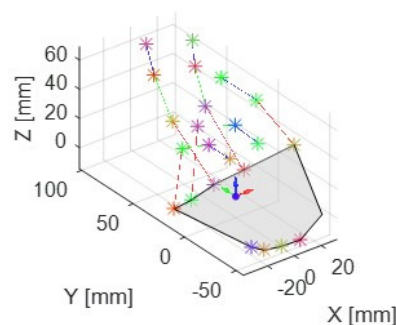


Figure 2. Leap Motion Controller connected to a computer that runs the Leap Motion Visualizer software showing the hands on top of the LMC camera [51].



(a)



(b)

Figure 3. Experimental setup (a) and hand model in the global coordinate system (b).

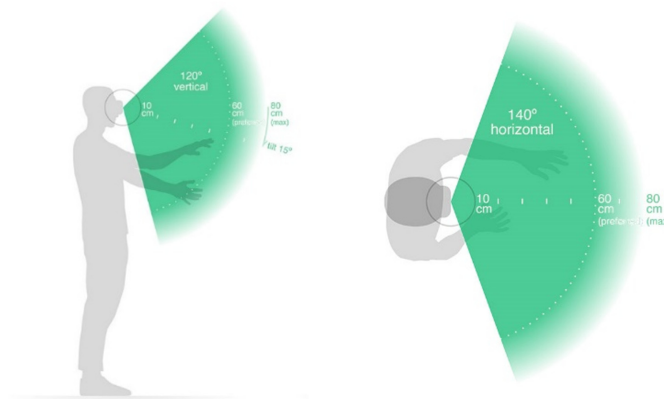


Figure 4. LMC's interaction zone [53].

Each subject attended one session of data collection, and six of the participants completed two sets of tasks while two of them only completed one due to time limitations. Each set of tasks contained eight randomly distributed tasks, and the order of the tasks in the two sets was different. The subjects were asked to rest for 45 s between tasks to avoid muscle fatigue. During each task, the subjects were seated on a regular office chair with back support. Each task was performed with the participant's dominant hand and was composed of static and dynamic phases. In the static phase, the participants were instructed to rest their forearms on a regular office desk to avoid tremor and hold an object, as listed in Table 1, for around 10 s, similar to how they would hold it in daily life. In the dynamic phase of the task, they were instructed to utilize the object over the entire range of motion that is usually performed in daily living at their own pace. Each dynamic task was repeated continuously 5 times without any rest intervals. Table 1 and Figure 5 demonstrate the ADL tasks.

Table 1. Dynamic tasks of the ADL dataset [54].

Object	Dynamic Task
Cup	Grabbing a cup from the table top and bringing it to mouth to pretend drinking from the cup and put it back on the table
Fork	Bringing pretended food from a paper plate on the table to the person's mouth
Key	Locking/unlocking a pretended door lock while holding a car key
Knife	Cutting a pretended stake by moving the knife back and forth
Nail Clipper	Holding a nail clipper and pressing/releasing its handles
Pen	Tracing one line of uppercase letter "A"s, with 4 randomly distributed font sizes
Spherical Doorknob *	Rotating a doorknob clockwise and counter clockwise
Spoon	Bringing pretended food from a paper plate on the table to the person's mouth

* A cup was used instead of a spherical doorknob and the participants were instructed to mimic the hand posture of holding a spherical doorknob.

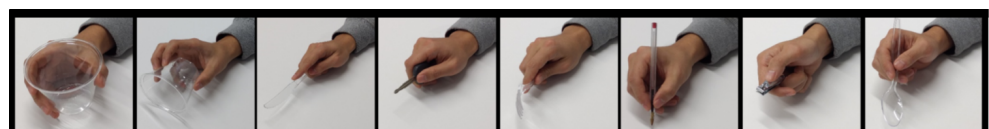


Figure 5. ADL tasks [54].

2.2. Preprocessing

The LMC provides the coordinates of hand joints and the palm center, as demonstrated in Figure 6, in 3-dimensional space. It also provides the coordinates of three orthonormal vectors at the palm center, which form the hand coordinate system (HCS), as shown in Figure 7. These coordinates are in units of millimeters with respect to the LMC frame of reference. The origin of the LMC's frame of reference is located at the top center position of the hardware, as presented in Figure 8. Therefore, while a participant performed a particular task, referred to as a trial hereafter, in each sample, i.e., each frame of the depth sensor, 84 coordinate values were recorded. The output of the LMC for each trial is a matrix of $n \times 84$, where n is the number of samples, i.e., the number of frames.

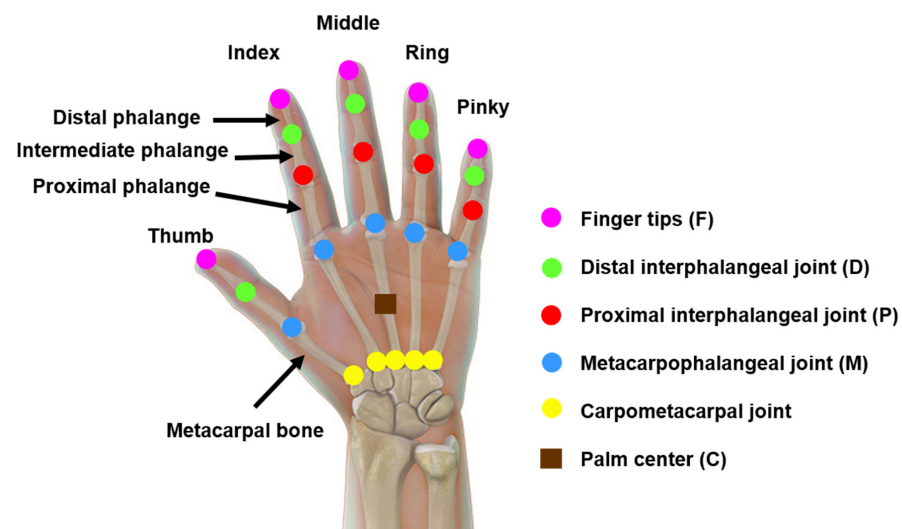


Figure 6. Hand joints and palm center [55].

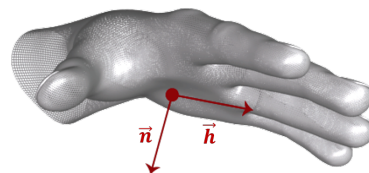


Figure 7. Hand coordinate system [56].

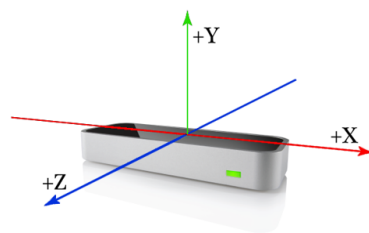


Figure 8. Leap Motion Controller frame of reference [57].

2.2.1. Change of Basis

The first preprocessing step was to transform LMC data from the LMC coordinate system to GCS using the Denavit–Hartenberg parameters [58] of the Cyton robot, since the LMC was rigidly attached to the end-effector of Cyton.

Once the LMC data had been transformed to GCS, the data were linearly translated into the hand palm center. Afterwards, by using a change of basis matrix at each frame, data were transferred from GCS to HCS based on Equation (1). In this equation, A is the change-of-basis matrix, or transition matrix, and its columns are the coordinates of the

basis vectors of HCS in the GCS at each frame [59]. $XHCS$ and $XGCS$ are the data matrices in HCS and GCS, respectively.

$$XHCS = inverse(A) * XGCS \quad (1)$$

During the trials, the hand grasps, i.e., the relative positions and orientations of the fingers and palm, did not change. In this work, the hand grasps were used for classifying different ADL tasks. Therefore, upper limb trajectories during the dynamic phase of the tasks, e.g., the entire-hand motions from plate to mouth while performing the “spoon” task, captured in the GCS needed to be removed. Transforming data from GCS to HCS eliminated gross hand motions and left the hand grasp information.

2.2.2. Filtering

At the next step, the transformed data were filtered using a median filter on a window size of 5 sampling points, i.e., 1/6 s.

2.3. Features and Classifiers

2.3.1. Feature Extraction

The choice of features used to represent the raw data can significantly affect the performance of the classification algorithms [60]. In this work, three groups of features, as presented in Table 2, were calculated for each trial and later combined for classification. The features are explained in detail in the following text.

Table 2. Feature categories.

Time-domain	Geometrical	AFA, ATD, DPUV, FHA, FTE, JA, NPTD
	Non-geometrical	MAV, RMS, VAR, WL
Frequency-domain	DFT	
Description of acronyms: Adjacent Fingertips Angle(AFA), Adjacent Tips Distance (ATD), Distal Phalanges Unit Vectors (DPUV), Fingertip- \vec{h} Angle (FHA), Fingertip Elevation (FTE), Joint Angle (JA), Normalized Palm-Tip Distance (NPTD), Mean Absolute Value (MAV), Root Mean Square (RMS), Variance (VAR), Waveform Length (WL), Discrete Fourier Transform (DFT)		

Geometrical features in the time domain

In order to compensate for different hand sizes, the features needed to be normalized. The geometrical features representing angles were divided by π , whereas the distance features were normalized to M . M is the accumulative Euclidean distance between the palm center and tip of the middle finger. At each sampling point, M was calculated by summation over the distance between the palm center and the metacarpophalangeal joint and the lengths of all three bones of the middle finger, as presented in Equation (2). Since there was less variation between participants’ hand grasps while performing the “cup” task, the coordinates of this task were used for the M calculation. The final length used for normalization was calculated by averaging M over the first 30 sampling points, i.e., the first second, of the first trial of the “cup” task.

$$M = |\vec{CM}| + |\vec{MP}| + |\vec{PD}| + |\vec{DF}| \quad (2)$$

1. Adjacent Fingertips Angle (AFA): This feature demonstrates the angle between every two adjacent fingertip vectors, which is the angle between the vectors from the palm center to the fingertips. The AFA is calculated by Equation (3), where F_i represents the fingertip location. This feature was normalized to the interval of [0, 1] by dividing the an-

gles by π . Lu et al. [61] achieved a classification accuracy of 74.9% using the combination of this feature and the hidden conditional neural field (HCNF) as the classifier.

$$AFA = \frac{\angle(F_i, F_{i+1})}{\pi} \quad i = 1, 2, \dots, 4 \quad (3)$$

2. Adjacent Tips Distance (ATD): This feature represents the Euclidean distance between every two adjacent fingertips and is calculated by Equation (4), in which F_i represents the fingertip location. There are four spaces between the five fingers of each hand, so there are four ATDs in each hand. This feature was normalized to the interval of $[0, 1]$ by dividing the calculated distances by M . Lu et al. [61] achieved an accuracy level of 74.9% by using the combination of this feature and HCNF.

$$ATD = \frac{|F_i - F_{i+1}|}{M} \quad i = 1, 2, \dots, 4 \quad (4)$$

3. Distal Phalanges Unit Vectors (DPUV) [62]: For each finger, the distal phalanges vector is defined as the vector from the distal interphalangeal joint to the fingertip, as presented in Figure 6. This feature was normalized by dividing by its norm.
4. Normalized Palm-Tip Distance (NPTD): This feature represents the Euclidean distance between the Palm Center and each fingertip. The NPTD is calculated by Equation (5) where F_i represents the fingertip location, and C is the location of the palm center. This feature was normalized to the interval $[0, 1]$ by dividing the distance by M . Lu et al. [61] achieved an accuracy level of 81.9% using the combination of this feature and HCNF, while Marin et al. [63] achieved an accuracy level of 76.1% using the combination of the Support Vector Machine (SVM) with the Radial Basis Function (RBF) kernel and Random Forest (RF) algorithms.

$$NPTD = \frac{|F_i - C|}{M} \quad i = 1, 2, \dots, 5 \quad (5)$$

5. Joint Angle (JA) [64,65]: This feature represents the angle between every two adjacent bones at the interphalangeal and metacarpophalangeal joints. For example, for the distal interphalangeal joint, θ is derived by Equation (6).

$$\theta = \arccos\left(\frac{\vec{DF} \cdot \vec{PD}}{|\vec{DF}| |\vec{PD}|}\right) \quad (6)$$

6. Fingertip- \vec{h} Angle (FHA): This feature determines the angle between the vector from the palm center to the projection of every fingertip on the palm plane and \vec{h} , which is the finger direction of the hand coordinate system, as presented in Figure 8. FHA is calculated by Equation (7), in which F_i^p is the projection of the F_i on the palm plane. The palm plane is a plane that is orthogonal to the vector \vec{n} and contains \vec{h} . By dividing the angles by π , this feature was normalized to the interval of $[0, 1]$. Lu et al. [61] and Marin et al. [63] achieved accuracy levels of 80.3% and 74.2% when classifying FHA features by HCNF and by using the combination of RBF-SVM with RF.

$$FHA = \frac{\angle(F_i^p - C, h)}{\pi} \quad i = 1, 2, \dots, 5 \quad (7)$$

7. Fingertip Elevation (FTE): Another geometrical feature is the fingertip elevation, which defines the fingertip distance from the palm plane. The FTE is calculated by Equation (8) in which "sgn" is the sign function, and \vec{n} is the normal vector to the palm plane. Like previous features, the F_i^p is the projection of the F_i on the palm plane. Lu et al. [61] achieved an accuracy level of 78.7% using the combination of this feature and HCNF, while Marin et al. [63] achieved an accuracy level of 73.1% when classifying FTE features by the combination of SVM with the RBF kernel and RF.

$$FTE = \frac{\text{sgn}\left(\left(F_i - F_i^p\right) \cdot \vec{n}\right) \left\|F_i - F_i^p\right\|}{M} \quad i = 1, 2, \dots, 5 \quad (8)$$

Non-geometrical features in the time domain

In order to compensate for the variations imposed by different participants' hand sizes, the filtered data were normalized to M , which is described in the "geometrical features in the time domain" section. All non-geometrical time-domain features were calculated over a sliding window with a size of 15 samples, which equals 0.5 s, with no overlap between the windows.

1. Mean Absolute Value (MAV): The MAV was calculated by taking an average of the absolute values of the signal's amplitude, using Equation (9). The MAV has shown promising results for classifying hand gestures [54,60,66,67].

$$MAV = \frac{1}{N} \sum_{n=1}^N |X_n| \quad (9)$$

2. Root Mean Square (RMS): similar to the MAV, the RMS feature represents the signal in an average sense. The RMS feature is calculated using Equation (10), where X_n is the sampling point and N is the number of samples in the moving window [60,68].

$$RMS = \sqrt{\frac{1}{N} \sum_{n=1}^N X_n^2} \quad (10)$$

3. Variance (VAR): The variance of a signal is related to the deviation of the sampling points from their average, \bar{x} and is calculated by Equation (11). The variance is the mean value of the square of these deviations [60].

$$VAR = \frac{1}{N-1} \sum_{n=1}^N (x_n - \bar{x})^2 \quad (11)$$

4. Waveform length (WL): The waveform length is derived by summation over the numerical derivative of the samples and is given by Equation (12) [60,68,69].

$$WL = \sum_{n=1}^{N-1} |X_{n+1} - X_n| \quad (12)$$

Frequency-domain features

Discrete Fourier Transform (DFT): Since the coordinates were transferred to HCS, it is a valid assumption to assume that the grasps, and therefore the joint coordinates, were constant through an entire task. Therefore, the DFT was used to transfer signals from the time domain to the frequency domain. `numpy.fft.fft` was used to extract DFT features based on Equation (13), where $W_N = e^{-j2\pi/N}$ [70].

$$\begin{cases} X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk} & k = 0, 1, \dots, N-1 \\ x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-nk} & n = 0, 1, \dots, N-1 \end{cases} \quad (13)$$

2.3.2. Classification

The data matrix for each feature was formed by concatenating the features from all trials of all the tasks. The size of the obtained matrix was $n \times m$, where n is the number of sampling points from all trials of all tasks and m is the number of feature components. Data matrices were standardized to have zero mean and unit variance per column before being fed to the machine learning algorithms.

The SVM is well-known to be a strong classifier for hand gestures [23,44,71–76]. It is a robust algorithm for high-dimensional datasets with smaller numbers of sampling points.

The SVM maps data into a higher dimensional space and separates classes using an optimal hyperplane. In this study, the scikit-learn library [77] was used to implement the SVM with a Radial Basis Function (RBF), and the parameters were determined heuristically [78].

Moreover, a Convolutional Neural Network (CNN) was implemented in PyTorch [79,80] for classifying the tasks. CNN and its variations have been shown to be efficient algorithms for hand gesture classification [81–84]. The proposed architecture of the CNN is illustrated in Figure 9. The CNN architecture is composed of three convolution layers and one linear layer. The three convolution layers have output channels of 16, 32, and 32 in sequential order, and each convolution layer consists of 2×2 filters with a stride of 1 and zero padding of 1. The Rectified Linear Unit (ReLU) activation function and batch normalization function were applied at the end of each convolution layer, and the maximum pooling function was applied at the end of the first and second layers. A fifty % dropout was implemented at the end of the fully connected layer, i.e., after the linear function in Figure 9. The learning rate, epoch, and batch size for training the CNN algorithm were set to 0.01, 20, and 40, respectively. The hyperparameters were determined experimentally.

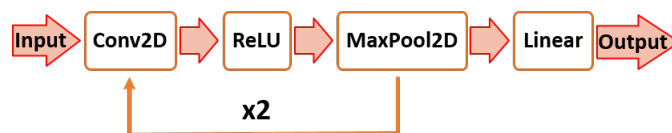


Figure 9. Proposed CNN architecture [54].

3. Results and Discussion

PCA dimensionality reduction, the adaptive learning rate for training the CNN algorithm, and different data filtering schemes were tested and were rejected as they were shown to be detrimental to the classification accuracy. The 5-fold cross validation performance metrics of the CNN and SVM algorithms in classifying the ADL tasks on the pure data, i.e., filtered data in HCS, as well as different combinations of features are presented in Tables 3 and 4, respectively. The precision, recall, and F1-score were calculated using the `sklearn.metrics.precision_recall_fscore_support` function by setting `average = 'macro'` to calculate these metrics for each class and report their average values.

Both algorithms were better at classifying some of the time-domain features when compared with their performance when classifying pure data. Among the time-domain, non-geometrical features, VAR and WL represent the data poorly, as they are calculated based on variations in the signal over time (Equations (11) and (12)). Since the data were transformed to HCS, the grasps, and consequently the coordinates of the joints, can be assumed to be constant over time. Therefore, VAR and WL are very similar in different tasks and cannot be used to discriminate tasks from each other. Similarly, DFT features can be assumed to represent the frequency decomposition of DC signals with different amplitudes. As a result, the interclass variability in this feature is not high enough to achieve a high classification accuracy.

Based on Tables 3 and 4, SVM and CNN have comparable accuracy levels when classifying geometrical features. However, SVM outperforms CNN when features are combined. This could be correlated to the ability of SVM to classify high-dimensional datasets, even when the number of samples is not proportionally high.

The classification accuracies achieved using the AFA and FTE features were lower than those achieved in a similar study [61]; however, the tasks classified in the two studies were very different. The ADL dataset includes many tasks in which the fingers are flexed while the hand holds an object. This minimizes the variation in AFA and FTE among the tasks. In addition, to have a meaningful comparison between the results of different studies, the inclusion or exclusion of gross hand motions in the classification should be taken into account. In the current analysis, information about the gross hand motions was removed from the data.

As demonstrated in Tables 3 and 4, ATD and JA are the best features for classifying the tasks using both algorithms. The ATD-CNN combination achieved a classification accuracy of over 99% and precision and recall values of over 97%. JA performed better when combined with the SVM algorithm. The JA-SVM combination achieved values of over 90% for both accuracy and precision and a recall of over 89%. Moreover, combining two or more time-domain features can improve the classification performance using the same classifiers. Confusion matrices for both classifiers and sample geometrical features achieved accuracy levels of over 70%, as presented in Figure 10. The uniform distribution of off-diagonal elements in these matrices shows that the algorithms were not overfitted to any of the classes using these features.

Table 3. Performance metrics for different combinations of features with the CNN as a classifier using 5-fold cross validation. All numbers are presented as percentage values (%). Different sets of features, based on Table 2, are shown in different colors.

Feature	CNN			
	Accuracy	Precision	Recall	F-Score
Pure data	63.5	50.5	40.2	41.2
MAV	85.1	80.5	80.1	80.2
RMS	84.1	78.1	77.8	77.9
VAR	34.8	32.9	23.5	23.3
WL	36.7	31.4	29.2	29.6
AFA	57.3	54.4	52.3	52.6
ATD	99.88	97.5	97.3	97.4
DPUV	72	68.8	68	68.3
FHA	70.2	66.1	65.3	65.5
FTE	41.5	29	25.4	24.5
JA	77.4	74.4	73.9	74.2
NPTD	71.5	68.4	67.6	67.9
DFT	58.4	53.4	50.4	51.4
JA+DPUV	80.4	77.1	76.6	76.8
JA+NPTD	78.8	74.3	73.7	74
MAV+RMS	84	79.5	78.9	79.2
MAV+JA+NPTD	88.4	83.8	83.6	83.7
MAV+JA+NPTD+DPUV	87.59	82.9	82.5	82.7

Table 4. Performance metrics for different combinations of features with the SVM as a classifier using 5-fold cross validation. All numbers are presented as percentage values (%). Different sets of features, based on Table 2, are shown in different colors.

Feature	SVM			
	Accuracy	Precision	Recall	F-Score
Pure data	68.9	70.5	67.3	68.2
MAV	79.5	81.1	77.9	78.99
RMS	76.3	78.4	74.7	75.8
VAR	24.6	61.1	21.8	20.3
WL	29.4	48.7	26.7	25.6
AFA	49.6	57.1	46.9	47.5
ATD	75.1	80.3	74.2	76.1
DPUV	79.3	79.2	78.3	78.7
FHA	64.2	66.2	62.5	63.3
FTE	30.8	50.8	29	30.4
JA	90.3	90.2	89.9	90
NPTD	79.3	79.2	78.3	78.6
DFT	52.4	77.6	50	54.3
JA+DPUV	94.7	94.4	94.4	94.4
JA+NPTD	92.3	92.2	91.9	92
MAV+RMS	79	80.5	77.5	78.4
MAV+JA+NPTD	92.5	92.3	91.9	92
MAV+JA+NPTD+DPUV	95.1	94.8	94.7	94.8

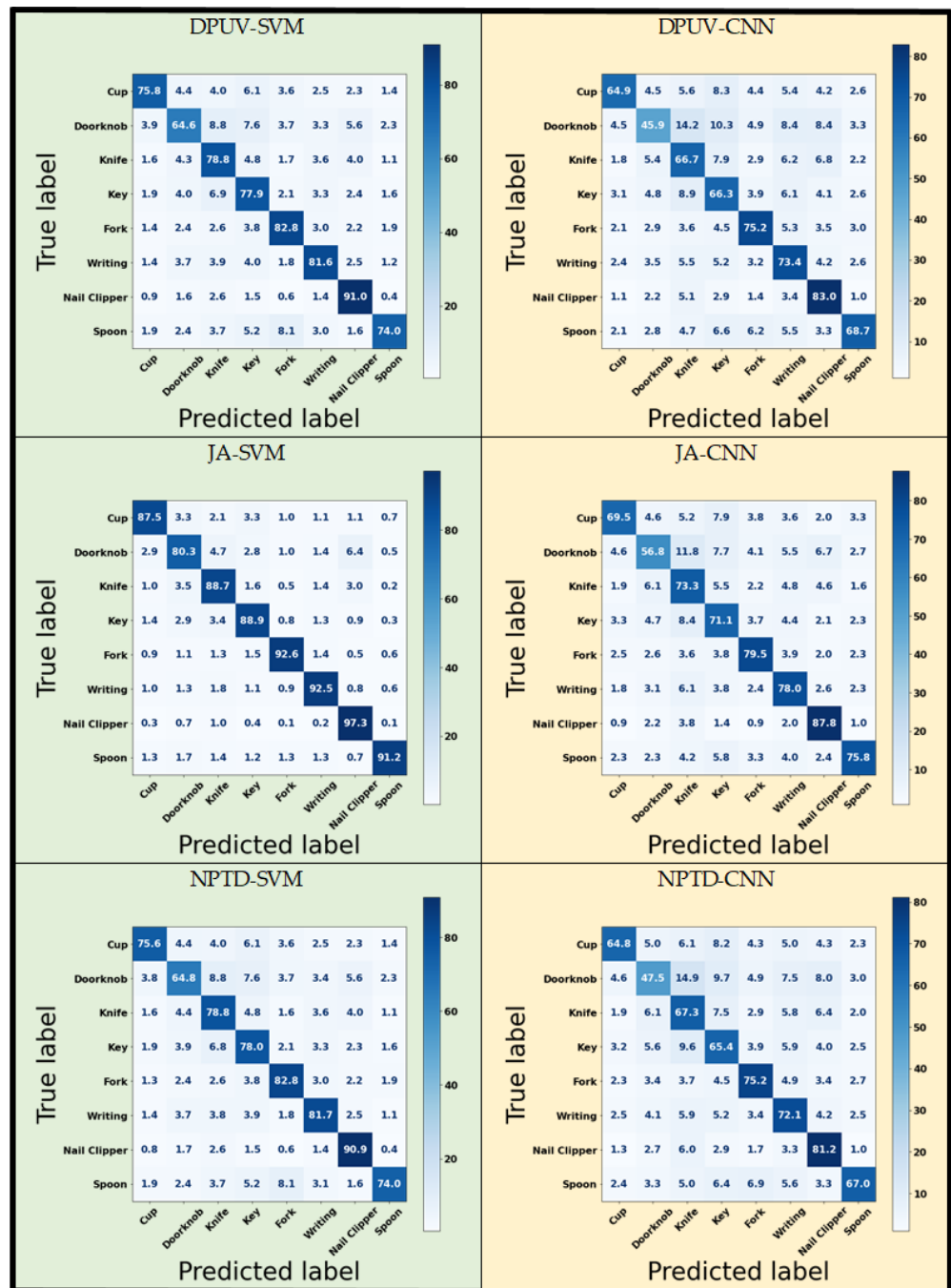


Figure 10. Confusion matrices for sample combinations of features and classifiers. All values were obtained through 5-fold cross validation and are presented as percentages (%).

4. Conclusions and Future Work

In this work, several classification systems were presented. These systems are made from the combination of a variety of time-domain and frequency-domain features with the SVM and CNN used as classifiers. The classification performance of the systems was tested on a proposed ADL dataset. The ADL dataset includes leap motion controller data collected from the upper limbs of healthy adults during the performance of eight common ADL tasks. To the best of authors' knowledge, this is the first ADL dataset collected by the LMC that includes both static hand grasps and dynamic hand motions of participants using real daily-life objects.

In this work, the data were transformed into HCS, so only the grasp information, and not the gross hand motions, were used for classification. A classification accuracy of over

99% and precision and recall values of over 97% were achieved by applying CNN on the “adjacent fingertips distance” feature. Eleven classification systems achieved a classification accuracy of over 80% with six achieving values of over 90% with high precision and recall values. Although the CNN and SVM had comparable performances for the individual features, for the combination of features, the SVM outperformed the CNN algorithm. From these observations, it can be deduced that the presented CNN algorithm may achieve a greater accuracy level if the size of the ADL dataset is increased.

The findings of this study pave the way for developing an ADL-assessment-metric in two ways. First, these findings can be immediately applied to evaluate a patient’s performance, and secondly, they can have long-term applications.

In the current study, a data analysis pipeline that takes LMC data from hand motions into account and outputs a classification accuracy to distinguish different ADL tasks was developed. Different preprocessing, feature extraction, and classification methods were tested on data collected from healthy adults to detect the best structure and parameters for the proposed pipeline. The developed pipeline can be set as a reference. Then, hand motion data from a neurological patient completing the same tasks with the same data collection setup can be fed into the reference pipeline to obtain the classification accuracy. The achieved accuracy indicates how close a patient’s hand motions are to the hand motions of the healthy population. This method enhances the assessment of the overall performance of a patient in a quantitative fashion. In addition, the acquired confusion matrix provides insight into the patient’s performance when completing each individual task.

As for the long-term applications, the features that achieve higher classification rates can be used for further analysis and for developing other metrics, as they represent different classes in a more distinguishable way. For instance, the distribution of these features in each ADL task among the healthy adults can be set as a reference metric. In this scenario, the location of a patient’s hand data in the reference distribution can be used to evaluate the patient’s performance and the rehabilitation progress. Greater analysis of the data from healthy adults as well as collection of the same data from neurological patients is required to complete this metric.

In conclusion, future work should be focused on three directions. Firstly, other classifiers should be investigated to increase the algorithm’s speed. Furthermore, the LMC data should be transformed back to the global coordinate system to include gross hand motions and implement time series algorithms for classification. Finally, the ADL dataset should be expanded by recruiting more healthy and neurological patients as participants to advance the proposed methodology further toward the development of a quantitative assessment method. Particularly, data from the neurological patients are crucial to generalize the findings of the current study for clinical applications.

Author Contributions: Conceptualization, H.S.; Data curation, H.S.; Formal analysis, H.S. and A.E.; Funding acquisition, T.K.; Methodology, H.S.; Software, H.S., A.E. and P.C.; Supervision, T.K.; Writing—original draft, H.S.; Writing—review & editing, A.E., P.C. and T.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Science Foundation, grant number 1502339.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki and approved by the Institutional Review Board of University of Illinois at Urbana-Champaign (protocol code 18529 and the date of approval was 17 July 2018).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Acknowledgments: The authors would like to thank Seung Byum Seo for providing technical advice on this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cramer, S.C.; Nelles, G.; Benson, R.R.; Kaplan, J.D.; Parker, R.A.; Kwong, K.K.; Kennedy, D.N.; Finklestein, S.P.; Rosen, B.R. A functional MRI study of subjects recovered from hemiparetic stroke. *Stroke* **1997**, *28*, 2518–2527. [[CrossRef](#)] [[PubMed](#)]
2. Hatem, S.M.; Saussez, G.; Della Faille, M.; Prist, V.; Zhang, X.; Dispa, D.; Bleyenheuft, Y. Rehabilitation of motor function after stroke: A multiple systematic review focused on techniques to stimulate upper extremity recovery. *Front. Hum. Neurosci.* **2016**, *10*, 442. [[CrossRef](#)] [[PubMed](#)]
3. Langhorne, P.; Bernhardt, J.; Kwakkel, G. Stroke rehabilitation. *Lancet* **2011**, *377*, 1693–1702. [[CrossRef](#)]
4. Available online: <http://www.strokeassociation.org/STROKEORG/AboutStroke/Impact-of-Stroke-Stroke-statistics/UCM/310728/Article.jsp#\WNPkhvnyAh> (accessed on 12 July 2017).
5. Duruoz, M.T. *Hand Function*; Springer: Berlin/Heidelberg, Germany, 2016. [[CrossRef](#)]
6. Demain, S.; Wiles, R.; Roberts, L.; McPherson, K. Recovery plateau following stroke: Fact or fiction? *Disabil. Rehabil.* **2006**, *28*, 815–821. [[CrossRef](#)]
7. Lennon, S. Physiotherapy practice in stroke rehabilitation: A survey. *Disabil. Rehabil.* **2003**, *25*, 455–461. [[CrossRef](#)]
8. Page, S.J.; Gater, D.R.; Bach-y Rita, P. Reconsidering the motor recovery plateau in stroke rehabilitation. *Arch. Phys. Med. Rehabil.* **2004**, *85*, 1377–1381. [[CrossRef](#)]
9. Matheus, K.; Dollar, A.M. Benchmarking grasping and manipulation: Properties of the objects of daily living. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 5020–5027. [[CrossRef](#)]
10. Katz, S. Assessing self-maintenance: Activities of daily living, mobility, and instrumental activities of daily living. *J. Am. Geriatr. Soc.* **1983**, *31*, 721–727. [[CrossRef](#)]
11. Dollar, A.M. Classifying human hand use and the activities of daily living. In *The Human Hand as an Inspiration for Robot Hand Development*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 201–216. [[CrossRef](#)]
12. Lawton, M.P.; Brody, E.M. Assessment of older people: Self-maintaining and instrumental activities of daily living. *Gerontologist* **1969**, *9*, 179–186. [[CrossRef](#)]
13. Mohammed, H.I.; Waleed, J.; Albawi, S. An Inclusive Survey of Machine Learning based Hand Gestures Recognition Systems in Recent Applications. In Proceedings of the IOP Conference Series: Materials Science and Engineering, Sanya, China, 12–14 November 2021; IOP Publishing: Bristol, UK, 2021; Volume 1076, p. 012047.
14. Allevard, T.; Benoit, E.; Foulloy, L. Hand posture recognition with the fuzzy glove. In *Modern Information Processing*; Elsevier: Amsterdam, The Netherlands, 2006; pp. 417–427. [[CrossRef](#)]
15. Garg, P.; Aggarwal, N.; Sofat, S. Vision based hand gesture recognition. *Int. J. Comput. Inf. Eng.* **2009**, *3*, 186–191.
16. Alonso, D.G.; Teyseyre, A.; Soria, A.; Berdun, L. Hand gesture recognition in real world scenarios using approximate string matching. *Multimed. Tools Appl.* **2020**, *79*, 20773–20794. [[CrossRef](#)]
17. Stingham Filho, I.A.; Gatto, B.B.; Pio, J.; Chen, E.N.; Junior, J.M.; Barboza, R. Gesture recognition using leap motion: A machine learning-based controller interface. In Proceedings of the Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2016 7th International Conference, Hammamet, Tunisia, 18–20 December 2016; IEEE: Piscataway, NJ, USA, 2016.
18. Chuan, C.H.; Regina, E.; Guardino, C. American sign language recognition using leap motion sensor. In Proceedings of the 2014 13th International Conference on Machine Learning and Applications, Detroit, MI, USA, 3–5 December 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 541–544.
19. Chong, T.W.; Lee, B.G. American sign language recognition using leap motion controller with machine learning approach. *Sensors* **2018**, *18*, 3554. [[CrossRef](#)] [[PubMed](#)]
20. Mohandes, M.; Aliyu, S.; Deriche, M. Arabic sign language recognition using the leap motion controller. In Proceedings of the 2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE), Istanbul, Turkey, 1–4 June 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 960–965. [[CrossRef](#)]
21. Hisham, B.; Hamouda, A. Arabic Static and Dynamic Gestures Recognition Using Leap Motion. *J. Comput. Sci.* **2017**, *13*, 337–354. [[CrossRef](#)]
22. Elons, A.; Ahmed, M.; Shedid, H.; Tolba, M. Arabic sign language recognition using leap motion sensor. In Proceedings of the 2014 9th International Conference on Computer Engineering & Systems (ICCES), Vancouver, BC, Canada, 22–24 August 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 368–373. [[CrossRef](#)]
23. Hisham, B.; Hamouda, A. Arabic sign language recognition using Ada-Boosting based on a leap motion controller. *Int. J. Inf. Technol.* **2021**, *13*, 1221–1234. [[CrossRef](#)]
24. Karthick, P.; Prathiba, N.; Rekha, V.; Thanalaxmi, S. Transforming Indian sign language into text using leap motion. *Int. J. Innov. Res. Sci. Eng. Technol.* **2014**, *3*, 5.
25. Kumar, P.; Gauba, H.; Roy, P.P.; Dogra, D.P. A multimodal framework for sensor based sign language recognition. *Neurocomputing* **2017**, *259*, 21–38. [[CrossRef](#)]
26. Kumar, P.; Saini, R.; Behera, S.K.; Dogra, D.P.; Roy, P.P. Real-time recognition of sign language gestures and air-writing using leap motion. In Proceedings of the 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA), Nagoya, Japan, 8–12 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 157–160. [[CrossRef](#)]

27. Zhi, D.; de Oliveira, T.E.A.; da Fonseca, V.P.; Petriu, E.M. Teaching a robot sign language using vision-based hand gesture recognition. In Proceedings of the 2018 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), Ottawa, ON, Canada, 12–14 June 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–6. [\[CrossRef\]](#)
28. Anwar, A.; Basuki, A.; Sigit, R.; Rahagiyanto, A.; Zikky, M. Feature extraction for Indonesian sign language (SIBI) using leap motion controller. In Proceedings of the 2017 21st International Computer Science and Engineering Conference (ICSEC), Bangkok, Thailand, 15–18 November 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1–5. [\[CrossRef\]](#)
29. Nájera, L.O.R.; Sánchez, M.L.; Serna, J.G.G.; Tapia, R.P.; Llanes, J.Y.A. Recognition of Mexican sign language through the leap motion controller. In Proceedings of the International Conference on Scientific Computing (CSC), Albuquerque, NM, USA, 10–12 October 2016; p. 147.
30. Simos, M.; Nikolaidis, N. Greek sign language alphabet recognition using the leap motion device. In Proceedings of the 9th Hellenic Conference on Artificial Intelligence, Thessaloniki, Greece, 18–20 May 2016; pp. 1–4.
31. Potter, L.E.; Araullo, J.; Carter, L. The leap motion controller: A view on sign language. In Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration, Adelaide, Australia, 25–29 November 2013; pp. 175–178.
32. Guzsvinecz, T.; Szucs, V.; Sik-Lanyi, C. Suitability of the Kinect sensor and leap motion controller—A literature review. *Sensors* **2019**, *19*, 1072. [\[CrossRef\]](#)
33. Castañeda, M.A.; Guerra, A.M.; Ferro, R. Analysis on the gamification and implementation of Leap Motion Controller in the IED Técnico industrial de Tocancipá. *Interact. Technol. Smart Educ.* **2018**, *15*, 155–164. [\[CrossRef\]](#)
34. Bassily, D.; Georgoulas, C.; Guettler, J.; Linner, T.; Bock, T. Intuitive and adaptive robotic arm manipulation using the leap motion controller. In Proceedings of the ISR/Robotik 2014; 41st International Symposium on Robotics, Munich, Germany, 2–3 June 2014; VDE: Offenbach, Germany, 2014; pp. 1–7.
35. Chen, S.; Ma, H.; Yang, C.; Fu, M. Hand gesture based robot control system using leap motion. In Proceedings of the International Conference on Intelligent Robotics and Applications, Portsmouth, UK, 24–27 August 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 581–591. [\[CrossRef\]](#)
36. Siddiqui, U.A.; Ullah, F.; Iqbal, A.; Khan, A.; Ullah, R.; Paracha, S.; Shahzad, H.; Kwak, K.S. Wearable-sensors-based platform for gesture recognition of autism spectrum disorder children using machine learning algorithms. *Sensors* **2021**, *21*, 3319. [\[CrossRef\]](#)
37. Ameer, S.; Khalifa, A.B.; Bouhleb, M.S. Hand-gesture-based touchless exploration of medical images with leap motion controller. In Proceedings of the 2020 17th International Multi-Conference on Systems, Signals & Devices (SSD), Marrakech, Morocco, 28–1 March 2017; IEEE: Piscataway, NJ, USA, 2020; pp. 6–11. [\[CrossRef\]](#)
38. Karashanov, A.; Manolova, A.; Neshov, N. Application for hand rehabilitation using leap motion sensor based on a gamification approach. *Int. J. Adv. Res. Sci. Eng* **2016**, *5*, 61–69.
39. Alimanova, M.; Borambayeva, S.; Kozhamzharova, D.; Kurmangaiyeva, N.; Ospanova, D.; Tyulepberdinova, G.; Gaziz, G.; Kassenkhan, A. Gamification of hand rehabilitation process using virtual reality tools: Using leap motion for hand rehabilitation. In Proceedings of the 2017 First IEEE International Conference on Robotic Computing (IRC), Taichung, Taiwan, 10–12 April 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 336–339. [\[CrossRef\]](#)
40. Wang, Z.r.; Wang, P.; Xing, L.; Mei, L.p.; Zhao, J.; Zhang, T. Leap Motion-based virtual reality training for improving motor functional recovery of upper limbs and neural reorganization in subacute stroke patients. *Neural Regen. Res.* **2017**, *12*, 1823. [\[CrossRef\]](#)
41. Li, W.J.; Hsieh, C.Y.; Lin, L.F.; Chu, W.C. Hand gesture recognition for post-stroke rehabilitation using leap motion. In Proceedings of the 2017 International Conference on Applied System Innovation (ICASI), Sapporo, Japan, 13–17 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 386–388. [\[CrossRef\]](#)
42. Škraba, A.; Koložvari, A.; Kofjač, D.; Stojanović, R. Wheelchair maneuvering using leap motion controller and cloud based speech control: Prototype realization. In Proceedings of the 2015 4th Mediterranean Conference on Embedded Computing (MECO), Budva, Montenegro, 14–18 June 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 391–394. [\[CrossRef\]](#)
43. Travaglini, T.; Swaney, P.; Weaver, K.D.; Webster, R., III. Initial experiments with the leap motion as a user interface in robotic endonasal surgery. In *Robotics and Mechatronics*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 171–179. [\[CrossRef\]](#)
44. Qi, W.; Ovr, S.E.; Li, Z.; Marzullo, A.; Song, R. Multi-sensor guided hand gesture recognition for a teleoperated robot using a recurrent neural network. *IEEE Robot. Autom. Lett.* **2021**, *6*, 6039–6045. [\[CrossRef\]](#)
45. Bachmann, D.; Weichert, F.; Rinkenauer, G. Review of three-dimensional human-computer interaction with focus on the leap motion controller. *Sensors* **2018**, *18*, 2194. [\[CrossRef\]](#) [\[PubMed\]](#)
46. Nogales, R.; Benalcázar, M. Real-time hand gesture recognition using the leap motion controller and machine learning. In Proceedings of the 2019 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Guayaquil, Ecuador, 11–15 November 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–7. [\[CrossRef\]](#)
47. Rekha, J.; Bhattacharya, J.; Majumder, S. Hand gesture recognition for sign language: A new hybrid approach. In Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCVC), Las Vegas, NV, USA, 18–21 July 2011; p. 1.
48. Rowson, J.; Yoxall, A. Hold, grasp, clutch or grab: Consumer grip choices during food container opening. *Appl. Ergon.* **2011**, *42*, 627–633. [\[CrossRef\]](#)

49. Cutkosky, M.R. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *IEEE Trans. Robot. Autom.* **1989**, *5*, 269–279. [[CrossRef](#)]
50. Available online: http://new.robai.com/assets/Cyton-Gamma-300-Arm-Specifications_2014.pdf (accessed on 12 July 2022).
51. Yu, N.; Xu, C.; Wang, K.; Yang, Z.; Liu, J. Gesture-based telemanipulation of a humanoid robot for home service tasks. In Proceedings of the 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), Shenyang, China, 8–12 June 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1923–1927. [[CrossRef](#)]
52. Available online: <https://www.ultraleap.com/product/leap-motion-controller/> (accessed on 12 July 2022).
53. Available online: <https://www.ultraleap.com/company/news/blog/how-hand-tracking-works/> (accessed on 12 July 2022).
54. Sharif, H.; Seo, S.B.; Kesavadas, T.K. Hand gesture recognition using surface electromyography. In Proceedings of the 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Montreal, QC, Canada, 20–24 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 682–685. [[CrossRef](#)]
55. Available online: <https://www.upperlimbclinics.co.uk/images/hand-anatomy-pic.jpg> (accessed on 12 July 2022).
56. Available online: https://developer-archive.leapmotion.com/documentation/python/devguide/Leap_Overview.html (accessed on 12 July 2022).
57. Available online: https://developer-archive.leapmotion.com/documentation/csharp/devguide/Leap_Coordinate_Mapping.html#:~:text=Leap%20Motion%20Coordinates,10cm%2C%20z%20%3D%20%2D10cm (accessed on 12 July 2022).
58. Craig, J.J. *Introduction to Robotics: Mechanics and Control*; Pearson Education: London, UK, 2005.
59. Change of Basis. Available online: <https://math.hmc.edu/calculus/hmc-mathematics-calculus-online-tutorials/linear-algebra/change-of-basis> (accessed on Day Month Year).
60. Patel, K. A review on feature extraction methods. *Int. J. Adv. Res. Electr. Electron. Instrum. Eng.* **2016**, *5*, 823–827. [[CrossRef](#)]
61. Lu, W.; Tong, Z.; Chu, J. Dynamic hand gesture recognition with leap motion controller. *IEEE Signal Process. Lett.* **2016**, *23*, 1188–1192. [[CrossRef](#)]
62. Yang, Q.; Ding, W.; Zhou, X.; Zhao, D.; Yan, S. Leap motion hand gesture recognition based on deep neural network. In Proceedings of the 2020 Chinese Control And Decision Conference (CCDC), Hefei, China, 22–24 August 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 2089–2093. [[CrossRef](#)]
63. Marin, G.; Dominio, F.; Zanuttigh, P. Hand gesture recognition with jointly calibrated leap motion and depth sensor. *Multimed. Tools Appl.* **2016**, *75*, 14991–15015. [[CrossRef](#)]
64. Avola, D.; Bernardi, M.; Cinque, L.; Foresti, G.L.; Massaroni, C. Exploiting recurrent neural networks and leap motion controller for the recognition of sign language and semaphoric hand gestures. *IEEE Trans. Multimed.* **2018**, *21*, 234–245. [[CrossRef](#)]
65. Fonk, R.; Schneeweiss, S.; Simon, U.; Engelhardt, L. Hand motion capture from a 3d leap motion controller for a musculoskeletal dynamic simulation. *Sensors* **2021**, *21*, 1199. [[CrossRef](#)]
66. Li, X.; Zhou, Z.; Liu, W.; Ji, M. Wireless sEMG-based identification in a virtual reality environment. *Microelectron. Reliab.* **2019**, *98*, 78–85. [[CrossRef](#)]
67. Zhang, Z.; Yang, K.; Qian, J.; Zhang, L. Real-time surface EMG pattern recognition for hand gestures based on an artificial neural network. *Sensors* **2019**, *19*, 3170. [[CrossRef](#)] [[PubMed](#)]
68. Khairuddin, I.M.; Sidek, S.N.; Majeed, A.P.A.; Razman, M.A.M.; Puzi, A.A.; Yusof, H.M. The classification of movement intention through machine learning models: the identification of significant time-domain EMG features. *Peerj Comput. Sci.* **2021**, *7*, e379. [[CrossRef](#)] [[PubMed](#)]
69. Abbaspour, S.; Lindén, M.; Gholamhosseini, H.; Naber, A.; Ortiz-Catalan, M. Evaluation of surface EMG-based recognition algorithms for decoding hand movements. *Med Biol. Eng. Comput.* **2020**, *58*, 83–100. [[CrossRef](#)] [[PubMed](#)]
70. Kehtarnavaz, N.; Mahotra, S. *Digital Signal Processing Laboratory: LabVIEW-Based FPGA Implementation*; Universal-Publishers: Irvine, CA, USA, 2010.
71. Kumar, B.; Manjunatha, M. Performance analysis of KNN, SVM and ANN techniques for gesture recognition system. *Indian J. Sci. Technol.* **2016**, *9*, 1–8. [[CrossRef](#)]
72. Huo, J.; Keung, K.L.; Lee, C.K.; Ng, H.Y. Hand Gesture Recognition with Augmented Reality and Leap Motion Controller. In Proceedings of the 2021 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, 13–16 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1015–1019. [[CrossRef](#)]
73. Li, F.; Li, Y.; Du, B.; Xu, H.; Xiong, H.; Chen, M. A gesture interaction system based on improved finger feature and WE-KNN. In Proceedings of the 2019 4th International Conference on Mathematics and Artificial Intelligence, Chengdu, China, 12–15 April 2019; pp. 39–43. [[CrossRef](#)]
74. Sumpeno, S.; Dharmayasa, I.G.A.; Nugroho, S.M.S.; Purwitasari, D. Immersive Hand Gesture for Virtual Museum using Leap Motion Sensor Based on K-Nearest Neighbor. In Proceedings of the 2019 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM), Surabaya, Indonesia, 17–18 November 2020; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6. [[CrossRef](#)]
75. Ding, I., Jr.; Hsieh, M.C. A hand gesture action-based emotion recognition system by 3D image sensor information derived from Leap Motion sensors for the specific group with restlessness emotion problems. *Microsyst. Technol.* **2020**, *28*, 1–13. [[CrossRef](#)]
76. Nogales, R.; Benalcázar, M. Real-Time Hand Gesture Recognition Using KNN-DTW and Leap Motion Controller. In Proceedings of the Conference on Information and Communication Technologies of Ecuador, Virtual, 17–19 June 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 91–103. [[CrossRef](#)]

77. Available online: <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html> (accessed on 30 June 2022).
78. Sha'Abani, M.; Fuad, N.; Jamal, N.; Ismail, M. kNN and SVM classification for EEG: A review. In *Lecture Notes in Electrical Engineering*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 555–565. [[CrossRef](#)]
79. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic Differentiation in Pytorch. 2017. Available online: <https://openreview.net/forum?id=BJJsrmfCZ> (accessed on 30 June 2022).
80. Available online: <https://pytorch.org/> (accessed on 30 June 2022).
81. Kritsis, K.; Kaliakatsos-Papakostas, M.; Katsouros, V.; Pikrakis, A. Deep convolutional and lstm neural network architectures on leap motion hand tracking data sequences. In Proceedings of the 2019 27th European Signal Processing Conference (EUSIPCO), A Coruna, Spain, 2–6 September 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–5. [[CrossRef](#)]
82. Naguri, C.R.; Bunesco, R.C. Recognition of dynamic hand gestures from 3D motion data using LSTM and CNN architectures. In Proceedings of the 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 18–21 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1130–1133. [[CrossRef](#)]
83. Lupinetti, K.; Ranieri, A.; Giannini, F.; Monti, M. 3d dynamic hand gestures recognition using the leap motion sensor and convolutional neural networks. In Proceedings of the International Conference on Augmented Reality, Virtual Reality and Computer Graphics, Lecce, Italy, 7–10 September 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 420–439. [[CrossRef](#)]
84. Ikram, A.; Liu, Y. Skeleton Based Dynamic Hand Gesture Recognition using LSTM and CNN. In Proceedings of the 2020 2nd International Conference on Image Processing and Machine Vision, Bangkok, Thailand, 5–7 August 2020; pp. 63–68. [[CrossRef](#)]