Towards arm tremor diagnosis using Kinect

Mohamed Elgendi[1], Flavien Picon[2] and Nadia Thalmann[2]

[1] Department of Computing Science, University of Alberta, Canada
[2] Institute of Media Innovation, Nanyang Technological University, Singapore

E-mail: moe.elgendi@gmail.com, piconflavien@ntu.edu.sg and nadiathalmann@ntu.edu.sg

Abstract
Many clinical studies have shown that the arm movement of patients with neurological injury is often slow. In this paper, the speed analysis of arm movement is presented, with the aim of evaluating the arm movement automatically using the Kinect camera. The arm movement seems simple at the first glance, but in reality it is very complex neural and biomechanical process that can be used for detecting a neurological disorder. This is a preliminary study investigated three different arm-movement speeds fast, medium and slow. Abnormality is associated with slowness in arm movement, while the fastness indicates healthy movement. With a sample size of 27 subjects, the proposed algorithm can classify the three different speed classes (slow, normal, and fast) with overall error 4.94 %. Moreover, the first 40% of the motion can be used to predict the speed type. Additional testing, on real patients with different hand and arm movement impairments is required to verify the effectiveness of the proposed approach.

Keywords: tremor movement disorder, arm movement disorder, Parkinson’s disease, Huntington’s chorea, Parkinson's disease, and cerebellar diseases

1. Introduction

According to the World Health Organization, Essential tremor (ET) affects an estimated 10 million people in the United States. ET is the most common adult movement disorder, and as much as 20 times more prevalent than Parkinson's disease.

The traditional view of ET is a progressive neurological disorder that causes involuntary shaking of particular parts of the body, usually the head and hands. However, the most recognizable feature is a tremor of the arms or hands that is apparent during voluntary movements such as eating and writing [1].

Although essential tremor is often mild, patients with severe tremor have difficulty performing many of their routine activities of daily living [2,3]. ET usually causes slowness in body-parts movement, which is more salient in the hand.

Slowness in arm movement is also common in many other disorders, such as Huntington's chorea [4], Parkinson's disease [5] and cerebellar diseases [6]. However the abnormality of arm movement varies from one disease to another.

Given the vast array of disorders associated with abnormal movements, rehabilitation community is challenged with receiving high quality evaluations at a reasonably affordable price.

Recently, Kinect offers an extremely inexpensive and effective tool for tracking body movements that is very promising to investigate tremor and slowness in arm movement.

Up to our knowledge, there are no studies which investigate the speed of arm-movement joints for detecting abnormality in arm movements using Kinect or any other depth cameras. However, several arm-movement recognition systems have considered the speed as a feature. Byung-Woo *et al*. [7] confirmed that arm-movement recognition is usually dependent on the trajectory of arm movement, and the position, speed, curvature are good features. Campbell *et al.* [8] investigated ten different features for arm-movement recognition using hidden Markov model (HMM). They indicated that speed features are superior to positional features. Yoon *et al*. [9] used the hand speed as an important feature for arm-movement recognition.

Other researchers estimated the speed of a arm-movement using an accelerometer, for example Rehm *et al*. [10] used the power of the accelerometer as feature to classify the arm movement into low and high speed.

In contrast to those studies, in this paper we systematically explore the speed of arm-movement joints, with the aim of improving the classification of the arm-movement speed. Our study is similar to Rehm's work, however we provide a device-free analysis of arm movement, explore the impact of different joints on the overall arm movement, and validate the system in a noisy environment.

In here we explore the speed of arm movement to be used in the evaluation of tremor and movement disorder.

This paper is structured as follows. The next section discusses the data collection. Section 3 demonstrates the methodology used in classifying speed types in arm movement, while Section 4 elaborates on the results. Finally the conclusion is covered in Section 5.

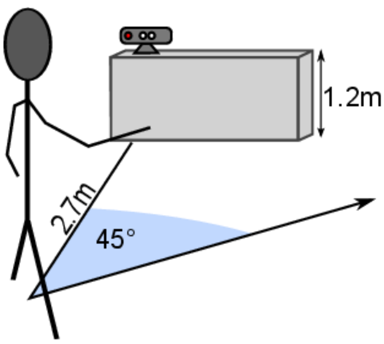


Figure 1: **Experimental Setup: the user is facing the camera with angle of 45° to the right of the sensor.** Every arm movement is recorded at fixed distance from the camera which is 2.7m where the camera is placed at height of 1.2m from above the floor.

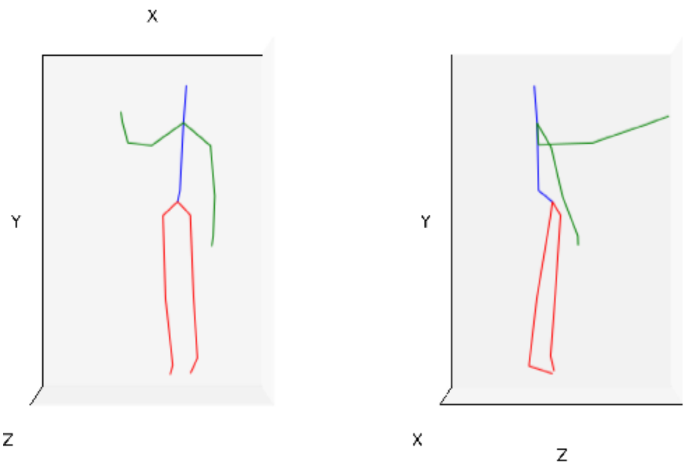


Figure 2: **Front and lateral view, of a subject, computed from the sensor data.** This plot represents the middle of the motion and was traced using Python 2.7 and the plotting module Matplotlib 1.1.0 [11]. The instantaneous velocity will be calculated using the *x, y, z* coordinates shown in the figure.

2. Data

Arm movement of 27 healthy volunteers (6 females and 21 males); two of them were left-handed, with a mean±SD age of 29.7±4.1, height of 172.9±9.3, arm length of 71.3±5.2. The motions were measured using the Kinect camera located 2.7 meters away from the subject at a height of 1.2 meters above the floor. The sampling frequency of the Kinect camera is 30 Hz.

During the experiment, the body of the user should be facing the sensor with an angle shift of 45° to the right of the Kinect sensor (cf. Fig. 1). The reason behind the 45° shift to the right is to prevent the arm joints from intersecting with the body joints, as shown in Fig. 2. This will generate reliable arm motion in order to study the impact of each joint of the arm on the overall speed of the right arm movement more precisely. These collected arm movements will be used as a benchmark for effective speed detection of an arm movement.

Measurements were taken while each subject is standing up with an initial position both arms extended along the body side. Then, the subject is asked to raise his right arm up in curved motion (cf. Fig. 2). Each subject performs three sets of trials: ‘slow’, ‘normal’, and ‘fast’; with five arm movements for each set. Therefore the number of recorded movements is 405 (27 subjects × 5 movements × 3 speeds).

The slow arm movement indicates Bradykinesia (a symptom of nervous system disorders, particularly Parkinson's disease) while the fast movement represents a healthy arm.

For the slow movement, the subject has been asked to raise his arm as if there is a heavy object in his arm that does not allow him to move normal.

Capturing the arm movement has been done manually. In other words, the subjects wait for a signal from the recording person to start their movement and then they maintain their arm up until they get a signal to come back to the initial position.

Each recorded joint has been played back to be checked and get annotated as one of three classes ‘slow’, ‘normal’, or ‘fast’. Two independent annotators have annotated the speed category of each recorded movement.

Annotation is a difficult task due to inter-annotator discrepancy, as the two annotators will never agree completely on what and how to annotate the speed class (‘slow’, ‘normal’, or ‘fast’) for each arm movement. Despite the annotation process being significantly time-consuming, discrepancies can be found in many records. Three cases will be discussed below to show how the discrepancies were adjudicated:

- Case 1:
  Annotator 1 agrees with Annotator 2 on all speed classes. Both annotators have no discrepancies, it is an optimal situation.
- Case 2:
  Both annotators agree on the speed class for most of the recorded arm movements.
- Case 3:
  Annotator 2 considered the speed class while Annotator 1 did not, and vice versa. This case has occurred few times in annotating the normal speed class, however, the subject will be asked to repeat the motion.

The annotations have been stored in a file to be compared automatically later with the speed features which will be discussed in the next section.

3. Methodology

The proposed arm-movement classification type algorithm consists of three main stages: pre-processing (resultant of coordinates as instantaneous velocity and lowpass filtering), feature extraction (calculating the first and second derivative and their mean and standard deviation) and classification (thresholding). The structure of the algorithm is shown in Figure 3.

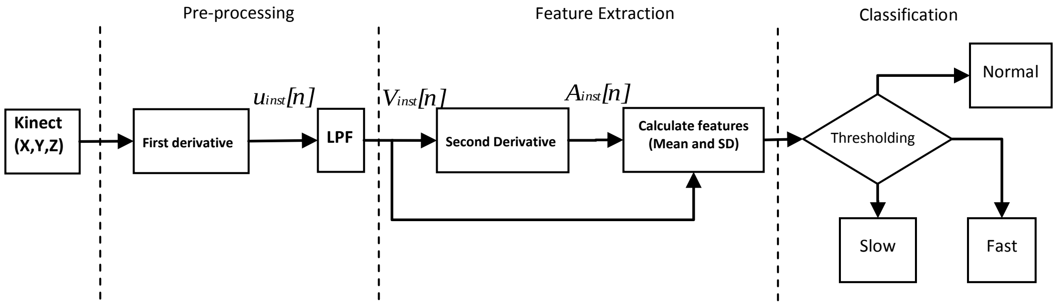


Figure 3 Flowchart for the arm-movement type classification. This is a time-domain algorithm that consists of three main stages: pre-processing, feature extraction and classification.

A. Pre-Processing

The Kinect API was used to record the position of a tracked subject with a frequency of 30 samples per second. Even though we focus mainly on the skeleton joints from the arm we chose to record the positions of all skeleton joints: center of gravity or legs movements could also be speed indicator. With 20 joints and 3 float values representing *x, y, z* positions for each joint, each motion frame is expressed as a vector of 60 floats.

The recorded joints cover all parts of the body but we focus mainly on the arm joints: shoulder, elbow, wrist and hand. Since the features only rely on the dynamics of the motion there is no differences in processing data from left or right arm. Therefore we could process data from the joints of the subject main arm (25 right-handed and 2 left-handed).

The 3D position (*x, y, z*) of a joint is expressed in the coordinate system of the Kinect and the unit is meter [12]. Again, the selected features rely on dynamics so our system is view-independent: we do not have to express the positions in the coordinate system of the subject's body.

The dynamic of each joint is computed using the variation of position of the joint over time. In a first step, each joint motion, sequence of 3D position, is replaced by the distance between each frame (cf. Eq. 1). In Figure 2, the *x, y, z* coordinates are the positions vectors of a particular joint that vary 0 to *n*; where *n* is the number of frames in a motion.

The instantaneous velocity of motion for a particular joint will be calculate as the resultant of *x, y, z* positions over all frames that represents a motion. The instantaneous velocity (*V*) in 3D motion is computed as follows:

$$u_{inst}[n] = \left.\frac{dx, y, z}{dt}\right|_{t=nT} = \frac{1}{T}\sqrt{(x[n]-x[n-1])^2 + (y[n]-y[n-1])^2 + (z[n]-z[n-1])^2}$$

Eq. 1

Where *T* is the sampling interval and equals the reciprocal of the sampling frequency, and *n* is the number of motion data points.

It is well-know that filtering helps in detecting certain events in biosignals. A low-pass filter has been applied. It is a first order Butterworth low pass filter with cutoff frequency of 2 Hz, used to reduce the generated noise from the environment and the small body movements. For sure the low frequencies are the main interest in this research as describes the arm movement compared to the high frequency.

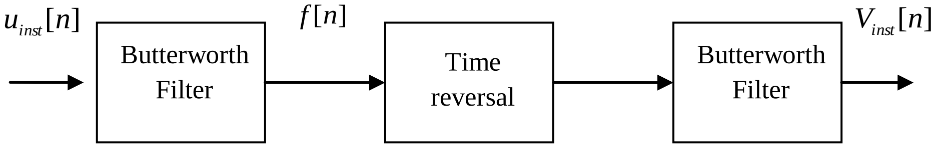


Figure 4 Demonstrating the zero-phase filtering over the resultant velocity of a motion

The bidirectional Butterworth filter is implemented as shown in Figure 4. The $V_{inst}[n]$ output will be a filtered version of $u_{inst}[n]$ with no phase distortion. The same Butterworth filter is used twice in this scheme: the time reversal step is a straight left–right flipping of the time-domain sequence, to produce zero-phase filtering, as follows:

$$f[n] = \sum_{k=0}^{N} b_k \, u_{inst}[n-k] - \sum_{k=1}^{N} a_k \, f[n-k] \qquad \text{Eq. 2}$$

$$V_{inst}[n] = \sum_{k=0}^{N} b_k \, f[n-k] - \sum_{k=1}^{N} a_k V_{inst}[n-k] \qquad \text{Eq. 3}$$

The first order filter has been selected to avoid over-smoothing the acquired motion and retain its meaningful property (cf. Figures 5 and 6). This has been done empirically to find a condition where the substantial part of the motion is preserved while sensor errors were strongly reduced.

We decided to record the raw data, i.e. without using the pre-defined filter provided in the Kinect SDK. By doing so, we have more control over the data analysis. We also can compare the effect of filtering on the classification rate.

B. Feature Extraction

Perhaps, before continuing the discussion of the joint signals, the calculated features over arm movements should be clarified first. In literature, the instantaneous velocity and acceleration have been used in diagnosing arm movements.

Almeida *et al.* [13] examined individuals with Parkinson's disease through the analysis of the upper-limbs movement at different movement frequencies, and with different external timing conditions using the instantaneous velocity. However, Helsen *et al.* [14] used the instantaneous velocity and acceleration to investigate the movements of the finger, elbow, and shoulder during a speeded aiming.

In here, two features will be investigated: the instantaneous velocity and acceleration. The mathematical definition of the instantaneous velocity (u_{inst}) in 3D motion before filtering is described in Eq. 1, while the instantaneous acceleration (A_{inst}) is defined in Eq. 4.

$$A_{inst}[n] = \left. \frac{dV_{inst}}{dt} \right|_{t=nT} = \frac{1}{T}(V_{inst}[n] - V_{inst}[n-1]) \qquad \text{Eq. 4}$$

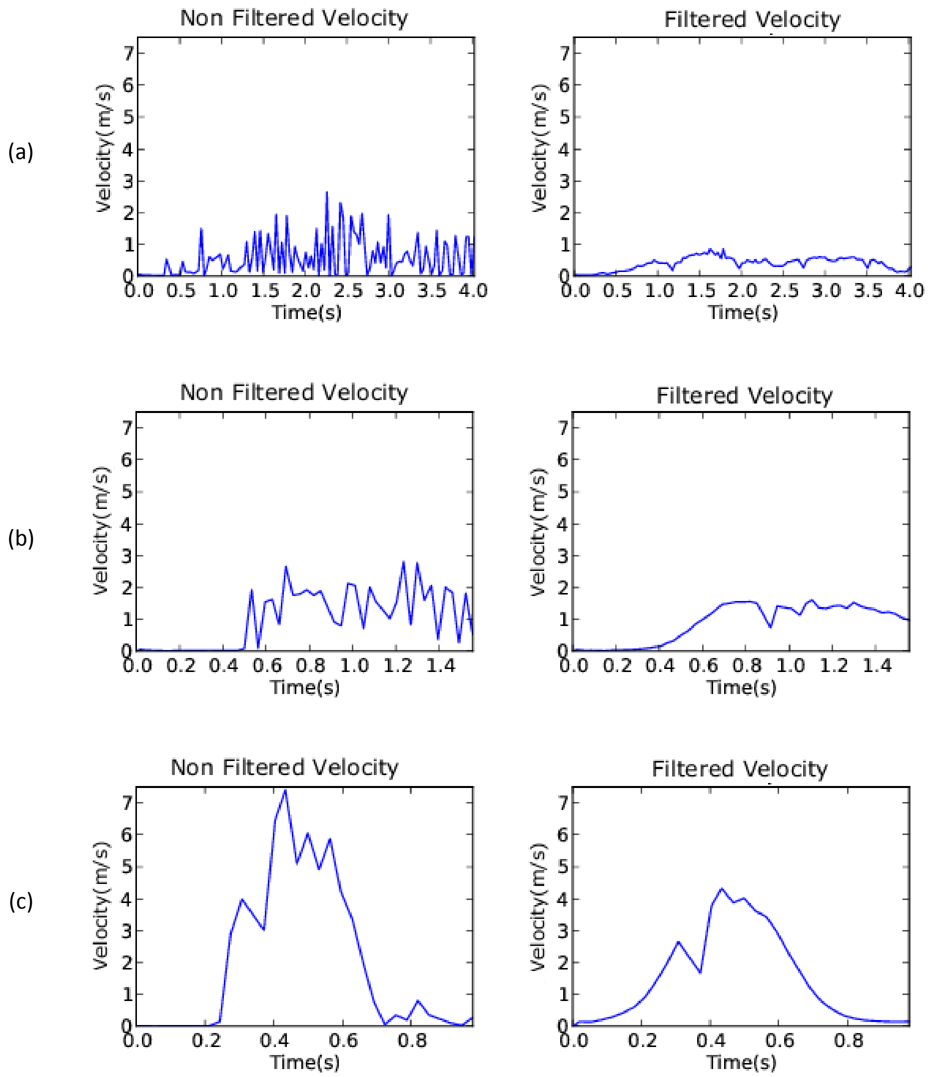


Figure 5: **Filtering effects on instantaneous velocity u_{inst} of right hand joint (a) slow, (b) medium, and (c) fast.** The low pass filter has been chosen to reduce low frequencies, i.e. sensor error or/and noise. The effects of filtering were also controlled by replaying recorded motions to check if they still displayed their main characteristics. Note: the sampling step for the filtered velocity (right side).

The skeleton positions contained in the raw signal are estimated at every frame; therefore it is very prone to errors that should affect the accuracy of the classification. Our first hypothesis is to observe a higher accuracy from filtered signal. Such filtering effect can also be observed on the instantaneous velocity data (cf. Fig. 5) and instantaneous acceleration data (cf. Fig. 6).

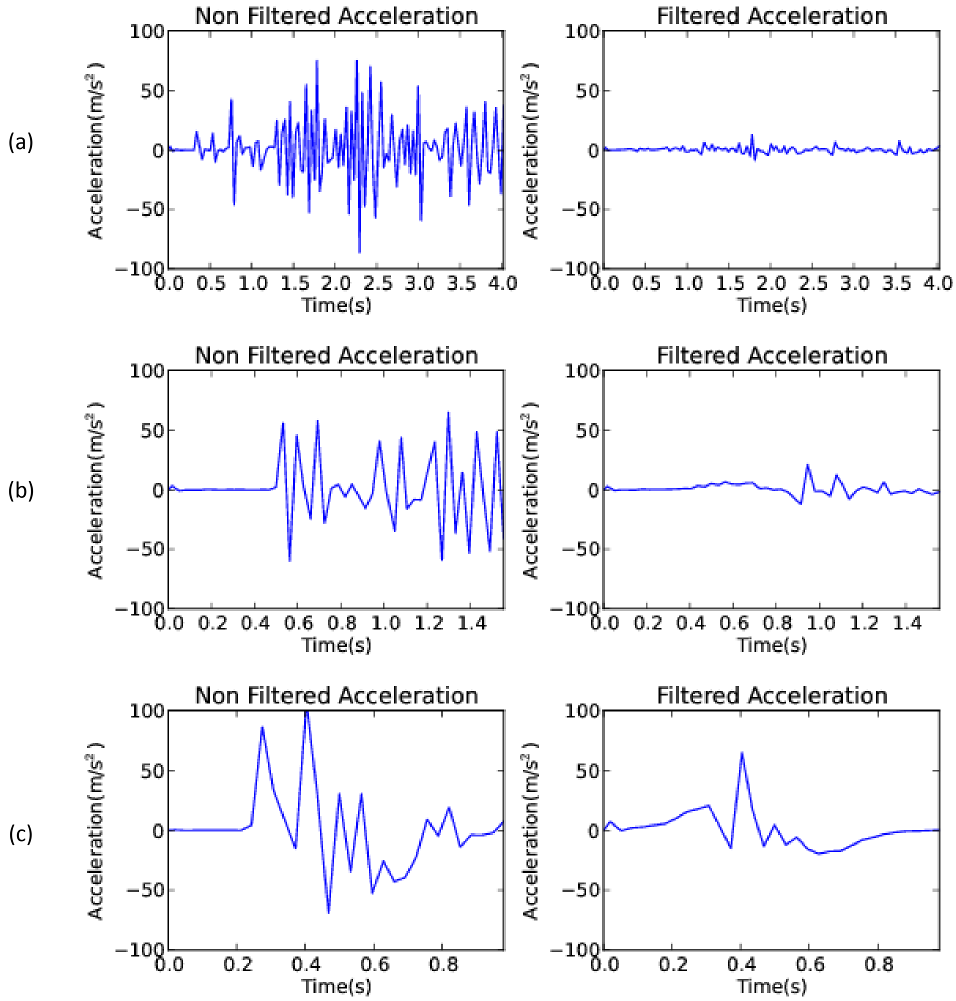


Figure 6: **Filtering effects on instantaneous acceleration A_{inst} of right hand joint (a) slow, (b) medium, and (c) fast.** It is clear that the Butterworth filter succeeded to reduce the high frequency noise in different speeds.

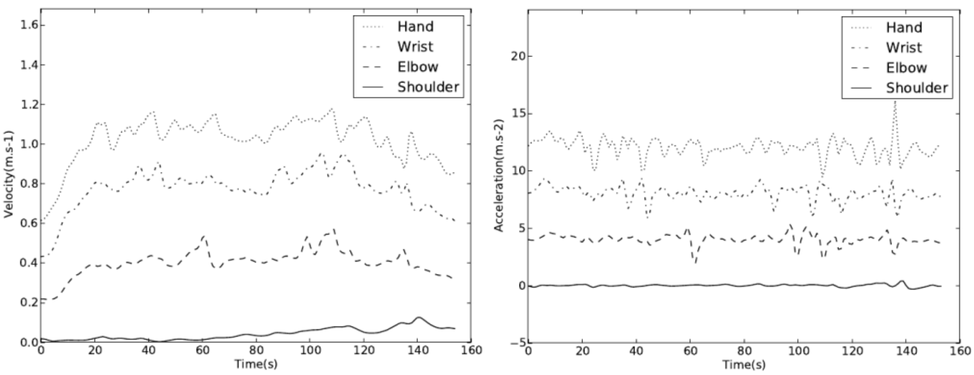


Figure 7: **Comparison of instantaneous velocity (at left) and acceleration (at right) of a slow motion for four joints: shoulder, elbow, wrist and hand of the right arm.** The plots are done for one motion of one subject. For a clearer graph, extra vertical space has been added between the plots, however the scale ratio has been preserved. From bottom to top are shoulder, elbow, wrist and hand respectively.

In our study, we computed the instantaneous velocity and instantaneous acceleration for each arm joint. Then, we extracted the following features: average (Mean) and standard deviation (SD).

Figure 7 demonstrates the signal shape of four different joints of a arm movement based on the instantaneous velocity and acceleration. This is particularly interesting as it confirms that joints of a same limb have the same dynamic, especially in the hand and wrist signals.

As the variance of the hand and wrist joint signals are quite higher compared to the elbow and shoulder signals, it is expected that the hand or the wrist signal would score higher accuracy in classifying the arm movements.

C. Classification

In this section, we check the linear velocity separability (based on the annotated files) over the calculated features in both filtered and non-filtered signals. Each motion contains a category parameter that was defined during the recording. We use this classification parameter to compare with the results of the feature classification. We perform inter subject classification: the process consists in 1) compute the feature value for each motion, 2) sort the motions based on the selected feature 3) compute the classification accuracy by counting the number of misclassified motions over total number of motions. In our case, we expect the 135 first motions to be 'fast' with black color, then the next 135 to be 'medium' with grey color and the last 135 to be 'slow' with white color (cf. Fig. 8). In a perfect situation, where all the motions collected accurately, the first part of Fig. 8 will be complete black, the middle part of the figure will be complete grey, while the last part will be complete white. Obviously it is not the case in here, we do have incorrect motions.

Figure 8 presents the motion ordered using the SD of velocity as a feature, the grey shade is used to show the initial classification (dark: fast, grey: medium and light grey: slow). The misplacement of the shades represents misclassification.

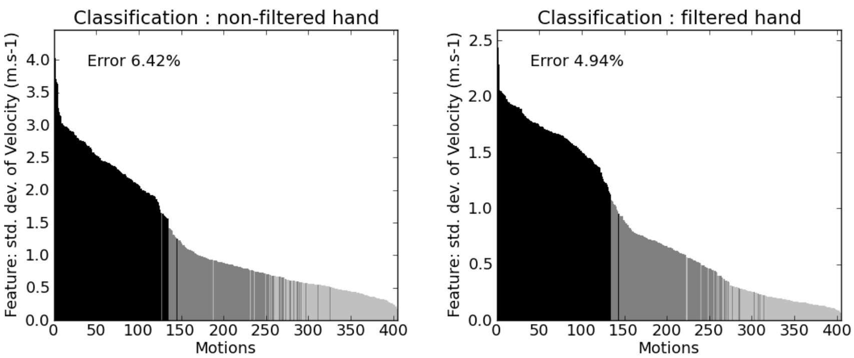
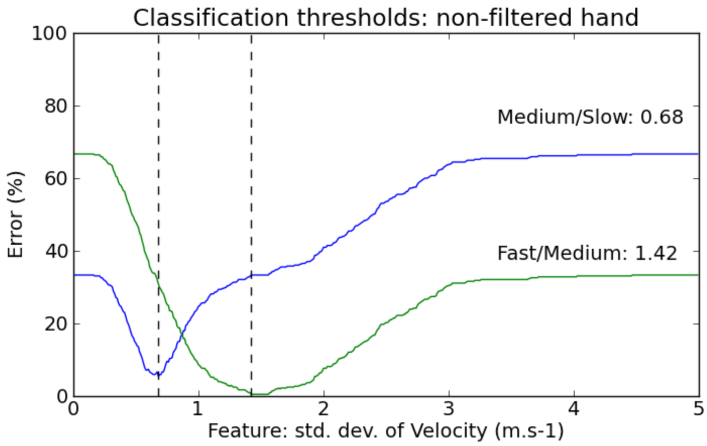


Figure 8: **Arm movements sorted in a descending order using the standard deviation of the instantaneous velocity of hand joint.** Motion velocities are split into three classes: dark for fast motion, grey for medium motion, white for slow motion. The line color represents the class/zone color. The non-filtered motions (left figure) shows a higher number of speed misclassifications compared to the filtered motions (right figure). Although the subjects have been asked to perform three distinctive motions (slow, medium, and fast), it did not prevent the overlapping (misclassification) between the speed classes.

(a)



(b)

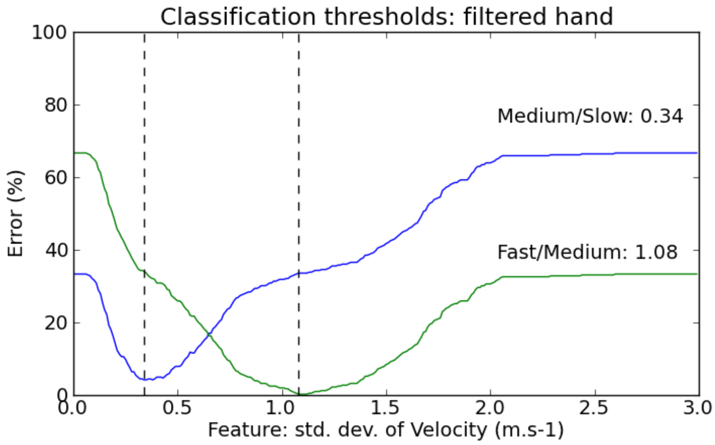


Figure 9: **Thresholds used for velocity classification: standard deviation of instantaneous velocity of hand joint (a) non-filtered (b) filtered.** The drop of the two curves show there is a clear separation between classes. The low error rate from the drops shows we achieve high separability. Slow/medium threshold appears in a small interval, while the value of the medium/fast threshold is not as precise. The two dashed lines refer to the drops of the two curves and their x-axis values are the thresholds. In non-filtered condition, the slow-medium threshold is 0.68 while the medium-fast threshold is 1.42, while the filtered condition, the slow-medium threshold is 0.34 while the medium-fast threshold is 1.08.

For automated speed-type detection, two classifiers have been run in order to specify the exact value of the thresholds. The first classifier is fast/medium against slow, while the second classifier is fast against medium/slow. This process generated two curves shown in Figure 10. Fortunately the two curves have two valleys reflect the thresholds that will be used for training the automatic speed detection. In non-filtered condition, the slow-medium threshold is 0.68 while the medium-fast threshold is 1.42, while the filtered condition, the slow-medium threshold is 0.34 while the medium-fast threshold is 1.08.

4. Results

Figure 9 shows an example of motion separability of one feature which is the SD of the instantaneous velocity of the right hand. As it is hard to distinguish the separability between the three speeds for filtered and non-filtered data, we ran Kruskal-Wallis and ANOVA tests to find out which approach is more classifiable.

It turns out, the filtered data scored less *p*-value (p=2.89x10e^{-77} in Kruskal-Wallis test and p=5.7x10^{-212} in ANOVA test) compared the non-filtered (p= 1.23x10^{-76} in

Kruskal-Wallis test and p=1.49x10^{-176} in ANOVA test). The very small p-value indicates that differences between the three speed classes are highly significant.

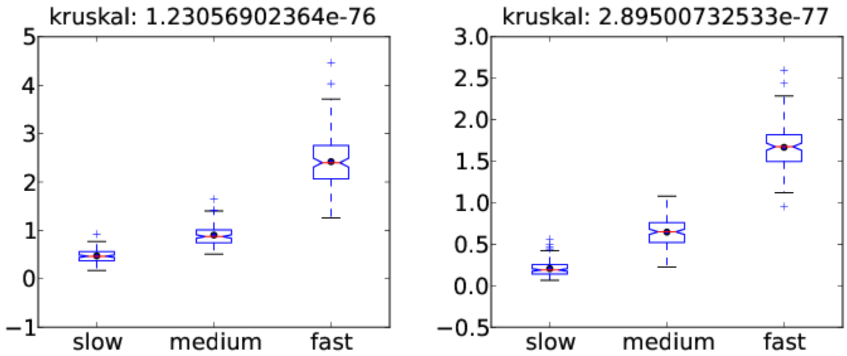


Figure 10: **Boxplot of the right arm, per speed class, for non-filtered (at left) and filtered arm-movement signal (at right). The calculated feature is the SD of the instantaneous velocity.** The p-values come from Kruskal-Wallis tests. The filtered data scored lower p-value compare to the raw data. Note the huge scale difference between non-filtered and filtered conditions.

The statistical analysis demonstrated that filtering the signals improves the separability between speed types.

In Table 1, as expected, the hand joint succeeded in classifying the speed types by scoring the lowest error rate (6.42% for non-filtered and 4.94% for filtered). This result confirms the observation shown in Figure 7 which is SD of the instantaneous velocity describes the motion in more detail.

Table 1 Classification Error in Percentage for Non-Filtered and Filtered arm-movement Signals. The SD of instantaneous velocity for a hand joint scored the lowest classification error rate: respectively 6.42% for non-filtered and 4.94% for filtered conditions.

	Hand				Wrist				Elbow				Shoulder			
	V_{inst}		A_{inst}		V_{inst}		A_{inst}		V_{inst}		A_{inst}		V_{inst}		A_{inst}	
	Mean (%)	SD (%)	Mean (%)	SD (%)	Mean (%)	SD (%)	Mean (%)	SD (%)	Mean (%)	SD (%)	Mean (%)	SD (%)	Mean (%)	SD (%)	Mean (%)	SD (%)
Non Filtered	8.40	6.42	60.74	30.86	10.37	12.35	58.02	41.98	11.36	17.53	51.60	34.32	27.41	31.36	38.52	33.58
Filtered	8.89	4.94	57.04	7.41	10.37	8.40	52.59	11.36	11.85	13.09	50.62	18.52	27.90	31.36	36.30	31.36

For the data, both the hand and wrist joints still are the most reliable joints for detecting speed in arm movement. It is interesting to note that features based on standard deviation perform better than those based on mean. We can also note that the instantaneous acceleration performed well in the case of the filtered hand-joint signal with classification error of 7.41%. Interestingly, the results of filtered and non-filtered hand-joint signal are relatively close. However, the filtered hand-joint signal scored a slightly lower classification error compared to the non-filtered signal.

The question that has been raised, what is the percentage of the motion contribute the most to the classification error. Therefore, we investigated different portions of the motion's signal using the same methodology that produced classification over a whole motion. The results of this investigation are shown in Figure 11.

The first 40% provides classification error 15.80±12.72%, while first 50% provides 5.43±8.32 classification error. This is an interesting observation as the first 40% and 50% of a motion provide low classification error and relatively close in terms of performance. Knowing this fact can lead to an effective prediction, which can be even done in realtime without waiting for the whole motion to be completed.

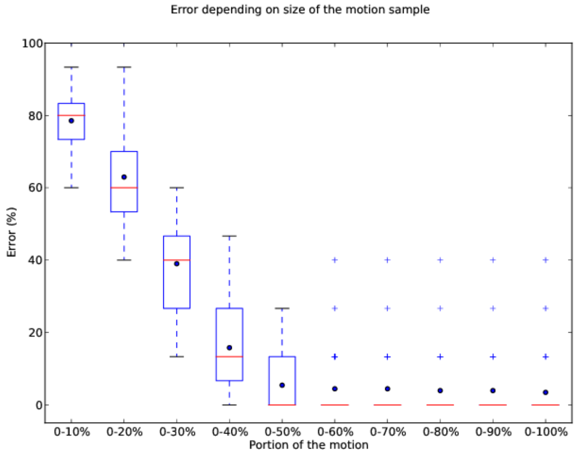


Figure 11 Classification error rate of the speed types based on the percentage of the whole motion used. The feature used here is the standard deviation of instantaneous velocity of the right-hand joint after applying lowpass filter. The portions 0-40% and 0-50% present a comparable error rate to the whole motion.

Well, another question can also be raised here, what portion within the motion's signal that contribute the most to the classification error. For example, what is the 10% of the motion's signal contains the most useful information to distinguish the speed types.

Figure 12 shows the error rate for a sequential 10% of the motion signals. It can be seen; the portions 20-30% and 70-80% of a whole motion provide lower error rate 28.39±15.14% and 27.41±13.74% respectively. This is intuitive as the beginning and ending of motion are phases where the subjects leave or reach a resting poses and thus do not yet characterize the motion. However, we need to know exactly which percentage specifically that can be used for analysis or/and prediction.

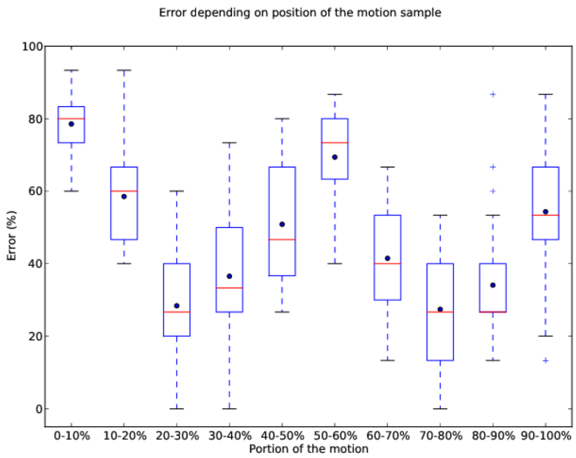


Figure 12 Classification error rate of the speed types based on a sequential 10% cuts of the whole motion. The feature used here is the standard deviation of instantaneous velocity of the right-hand joint after applying lowpass filter. The cuts 20-30% and 70-80% scored lower error rate.

5. Conclusion

In this paper we presented a speed analysis of arm movement. Results show that: 1) the instantaneous velocity provides more reliable classification compared to the instantaneous acceleration, 2) the standard deviation is slightly better than average, and 3) the hand joint is the most efficient joints for speed detection in an arm motion. Moreover, we can improve the accuracy by applying a lowpass filter.

The filtered instantaneous velocity scored 4.94% error rate in detecting the speed type over 405 motions. Moreover, the first 40% provides a classification error of 15.80±12.72%, which is relatively close performance to the whole motion, can be used for predicting the speed type in realtime. Furthermore, the most important 10% of a whole motion is 20-30% a whole motion.

The results are promising and this approach can be impeded in a human-computer-interaction system for interactive tremor diagnosis, specifically measuring hand-related disability and improvement.

In our approach we asked healthy subjects to show a faked abnormality by moving slowly, however testing this approach on patients with Parkinson disease or any hand tremors remains for future work.

Acknowledgements

This research, which is carried out at BeingThere Centre, is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

References

1. Benito-León J, Louis ED (2007) Clinical update: diagnosis and treatment of essential tremor. The Lancet 369: 1152-1154.
2. Bain P, Mally J, Gresty M, Findley L (1993) Assessing the impact of essential tremor on upper limb function. Journal of Neurology 241: 54-61.
3. Louis ED, Barnes L, Albert SM, Cote L, Schneier FR, et al. (2001) Correlates of functional disability in essential tremor. Movement Disorders 16: 914-920.
4. Bittenbender Jb QF (1962) RIgid and akinetic forms of huntington's chorea. Archives of Neurology 7: 275-288.
5. Jahanshahi M, Jenkins IH, Brown RG, Marsden CD, Passingham RE, et al. (1995) Self-initiated versus externally triggered movements: I. An investigation using measurement of regional cerebral blood flow with PET and movement-related potentials in normal and Parkinson's disease subjects. Brain 118: 913-933.
6. Deuschl G, Wenzelburger R, Löffler K, Raethjen J, Stolze H (2000) Essential tremor and cerebellar dysfunction Clinical and kinematic analysis of intention tremor. Brain 123: 1568-1580.
7. Byung-Woo M, Ho-Sub Y, Jung S, Yun-Mo Y, Toshiaki E. Hand gesture recognition using hidden Markov models; 1997 12-15 Oct 1997. pp. 4232-4235 vol.4235.
8. Campbell LW, Becker DA, Azarbayejani A, Bobick AF, Pentland A. Invariant features for 3-D gesture recognition; 1996 14-16 Oct 1996. pp. 157-162.
9. Yoon H-S, Soh J, Bae YJ, Seung Yang H (2001) Hand gesture recognition using combined features of location, angle and velocity. Pattern Recognition 34: 1491-1501.
10. Rehm M, Bee N, Andre E (2008) Wave like an egyptian - accelerometer based gesture recognition for culture specific interactions. BCS-HCI '08 Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction. pp. 13-22.
11. Hunter JD (2007) Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering 9: 90-95.
12. Microsoft MSDN Library (2012) Coordinate Spaces.
13. Almeida QJ, Wishart LR, Lee TD (2002) Bimanual coordination deficits with Parkinson's disease: The influence of movement speed and external cueing. Movement Disorders 17: 30-37.

14. Helsen WF, Elliott D, Starkes JL, Ricker KL (2000) Coupling of Eye, Finger, Elbow, and Shoulder Movements During Manual Aiming. Journal of Motor Behavior 32: 241-248.