

# PLAYING VIRTUAL MUSICAL DRUMS BY MEMS 3D ACCELEROMETER SENSOR DATA AND MACHINE LEARNING

Shaikh Farhad Hossain, Kazuhisa Hirose, Shigehiko Kanaya  
and Md. Altaf-Ul-Amin

Computational Systems Biology Lab, Graduate School of Information Science,  
Nara Institute of Science and Technology (NAIST), 8916-5,  
Takayama, Ikoma, Nara 630-0192, Japan

## ABSTRACT

*In our life, music is a vital entertainment part whose important elements are musical instruments. Foreexample, the acoustic drum plays a vital role when a song is sung. With the modern era, the style of the musical instruments is changing by keeping identical tune such as an electronic drum. In this work, we have developed "Virtual Musical Drums" by the combination of MEMS 3D accelerometer sensor data and machine learning. Machine learning is spreading in all arena of AI for problem-solving and the MEMS sensor is converting the large physical system to a smaller system. In this work, we have designed eight virtual drums for two sensors. We have found a 91.42% detection accuracy within the simulation environment and an 88.20% detection accuracy within the real-time environment with 20% windows overlapping. Although system detection accuracy satisfaction but the virtual drum sound was nonrealistic. Therefore, we implemented a 'multiple hit detection within a fixed interval, sound intensity calibration and sound tune parallel processing' and select 'virtual musical drums sound files' based on acoustic drum sound pattern and duration. Finally, we completed our "Playing Virtual Musical Drums" and played the virtual drum successfully like an acoustic drum. This work has shown a different application of MEMS sensor and machine learning. It shows a different application of data, sensor and machine learning as music entertainment with high accuracy.*

## KEYWORDS

*Virtual musical drum, MEMS, SHIMMER, support vector machines (SVM) and k-Nearest Neighbors (kNN)*

## 1. INTRODUCTION

Music is an important part of our lives to keep the mind fresh. The important supporting elements of music are pitch, dynamics, rhythm and presenting performance, while a song is sung with musical instruments that are played by musicians. Music has many different instruments and these can include piano, drum, guitar, ukulele, saxophone, violin, trumpet, cello, clarinet, and accordion. The drum is the most vital musical instrument and a member of the percussion group. The acoustic drum is played by the stick. Nowadays, an electronic drum, also known as digital drums or electric drums are available in the market and are used as an alternative to the acoustic drum kit. This modern electronic musical instrument is very popular which gives the same flavour of the acoustic drum. Now, it's the evolution of machine learning and micro electromechanical systems (MEMS) sensor. Use of machine learning is spreading in all areas of AI problem-solving which are doing our life easier. On the other hand, the MEMS sensor has been converted from a large physical system to a small physical system by keeping the same

accuracy. Machine learning and MEMS both are advanced technology. Using the combination of these two advanced inventions with Human Computer Interaction (HCI) system, we have tried to develop a musical system called "Playing Virtual Musical Drums" in which the user will be able to play drum music virtually by the movement of his hand posture (Figure 1)

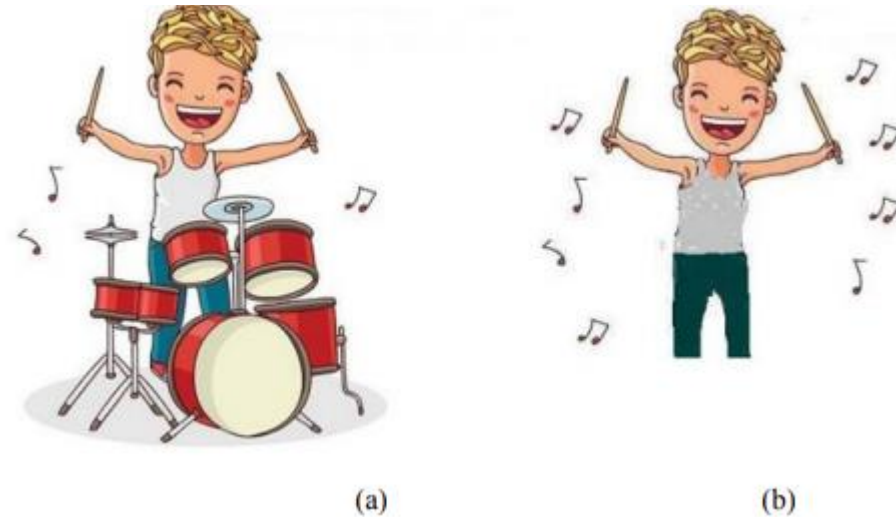


Fig. 1. (a). Real musical instruments (b). Virtual musical instruments playing

Major challenges in 3D gestural interface system are:

1. Sensor data capturing, buffering and monitoring data stream for training and classification.
2. Matching data vector with the specific pattern for detecting the predefined patterns.
3. 3D gesture recognizer algorithm evaluation.
4. In real-time, reaction time or latency evaluation whenever there is a match in the pattern.

By solving these major challenges, we have successfully developed "Playing Virtual Musical Drums" system that will assist the drum lover to play the virtual drum by retaining the same feel of these acoustic musical instruments. To purchase any musical instrument set, you need to pay a high cost. Moreover, larger instruments are also a problem to carry everywhere. In present, smartphone music applications are getting narrow to the user because it needs only finger movement. By using these music application, user don't get the real interest from this application, as a result, young generation feel boring. Compared with traditional conventional musical instruments, the system will give real flavor to the drum player because this work has been provided a state-of-the-art gesture tracking and recognition method by using advanced technology. In the modern era, everything is becoming automated and virtual, music needs to remain up-to-date as well.

## 2. BACKGROUND

Joseph et al. develop 3D hands gesture and body natural posture with computers interaction for a variety of different applications including, musical instrument, video games, training and simulation, and movement-based disease detect. They show a way to accurately analyse raw data and to 3D gestures correctly recognize [1]. Berman et al. constructed a gesture recognition system

(GRS) that was comprised of the sensor (gesture-capture device), motion capture (tracking), feature extraction and classification algorithm. He recommends that the determination of the context use, sensor stimulus, and sensor platform are the major preliminary designing issues in GRSs [2]. Chen et al. developed a 'Finexus' system by the use of magnetic sensing that tracked the precise motions of multiple fingertips. This system is useable for head-mounted displays during virtual reality and human input tasks, such as writing in the air. He reported that the average accuracy 'Finexus' system is better, as compared to results from an optical tracker [3]. Truong, Hoang, et al. proposed a 3D printed wristband system by including of wireless communication module (BLE), capacitive sensors arrays in a flexible circuit board and a low-cost micro-controller unit. To recognize and detect users' hand gestures, the wristband traces the changes in capacitance from multiple capacitive sensors [4]. Kang Li et al. designed and validated a glove-based multi-finger-motion capture wearable device, specifically focusing on the improvement of a new optical linear encoder (OLE). The OLE is compact, low-power consumption and lightweight. It showed that digital output is accurate and has high linearity [5]. Brown et al. presented an effective interface for gestural music interaction by using camera based motion tracking and leap motion optical sensor. The interface shows a more powerful tracking precision over a Leap Motion positioned on a table top [6]. Nymoen et al. invented a controller called MYO which consists of a 3D accelerometer, 3D gyroscope, a magnetometer and eight electromyography (EMG) sensors that an inertial measurement unit (IMU) and measure muscle tension. The purpose of this wearable device will be used for new musical instruments [7]. To play real-world musical instruments, Jeffrey et al. designed and implemented a virtual reality interface. The system has three main modules: the instrument robot, the webserver and the user interface. The user interface displays playable instruments as a virtual environment and translates user hand gestures. Then HTTP server will send translated activity to the webserver [8]. By analyzing the musical gesture and sound control, Caramiaux et al. provided a way how to use regression and classification for playing virtual music [9]. Torre et al. designed a new musical glove called "a live performance approach" which is a highly complex system in technical issues, sociological, musicological, psychological and anthropological. He presented a new approach to music performance often technologically mediated while outputting a great deal of new digital instruments [10].

Micro electro-mechanical systems (MEMS) technology for sensors provides huge advantage as it smaller and low power consumption. Wearable sensors have advantages that they are generally cheap, easy to carry and can be wear at any body location [10]. Due to the reliability, the MEMS sensor has occupied the medical sector for living independently especially older people and patients [11]. It allows the person to live normally in their own home ensuring that a caregiver or medical practitioner is taking care of his bodily health [12]. MEMS has a variety of applications in the modern era. So using MEMS sensor and machine learning, we have tried to show a different application of sensors where a new generation can be introduced within a new music playing technology. The work will also show that data is not only for information, but it can also keep a vital role in entertainment.

### **3. METHODOLOGY**

Below we have discussed the step by step procedure that we adopted for completing "Playing Virtual Musical Drums".

#### **A. Sensor type, number and dimensions selection**

Many different Micro-electromechanical systems (MEMS) sensors like gyroscope, magnetometer, accelerometer, encompass, pressure, and IR are available in the market. According to operation and application, we need to choose an accurate sensor. Based on the result analysis and observational analysis, playing the virtual drum using the MEMS accelerometer has

produced a good result. Using gyroscope or magnetometer (or both) is also possible to play the drum. In that case, if we use the individual sensor like gyroscopes or magnetometer, it shows a low percentage of accuracy, precision and recall to detect virtual drum. The combination of accelerometer and gyroscope, or gyroscopes and magnetometer shows a high percentage of accuracy, precision and recall. A gyroscope is needed for angledetection and an accelerometer is needed for motion velocity. The combination of the twosensors shows better virtual drum detection but it's a little bit costly. If we use the accelerometer with its 3 dimensions, it shows the same results like combination of accelerometer and gyroscope. So we can use only 3D accelerometer, instead of resultant accelerometer and gyroscope. The 3D accelerometer is providing both requirements of resultant accelerometer and 3 axis value which act like a gyroscope. For this study, we have selected a single MEMS 3D accelerometer for playing the virtual drum (Figure 2).

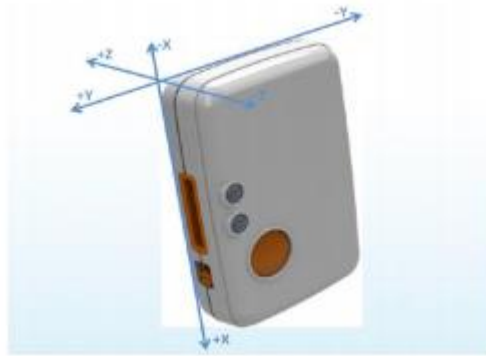


Fig. 2. Accelerometer and 3D coordinate systems

#### B. Sensor sampling frequency and sample number selection

The recognition of hand posture and virtual drum position is one of the most important tasks of this study. Setting a high sampling frequency means high accuracy posture recognition, but at the same time, it's a computational burden and sensor battery life. Sometimes the highfrequency setting of the sensor within a real-time environment creates a delayed response. We have used the SHIMMER (Sensing Health with Intelligence, Modularity, Mobility and Experimental Reusability) brand sensor [13]. In SHIMMER has 10 - 1000 Hz sampling frequency setting configuration option. Sequentially, 10Hz, 50Hz, 100Hz, 500Hz sampling frequency was set and collected train and test data. As we said earlier, high sampling frequency means high accuracy posture recognition. Based on observation and hypothesis, we have seen that 10 Hz shows a poor (56%) detection accuracy, 50Hz shows a good 84% detection accuracy, 100Hz shows a better (91%) detection accuracy and 500Hz shows better (95%) detection accuracy. Primarily, we have selected 50 Hz and 100Hz to detect hand posture recognition to play the virtual drum. Based on our natural drum playing duration, a 2.56s and 1.28s drum-hit were considered. A 2.56 second hit means 128 sampled data for 50Hz and 256 sampled data for a 100Hz sampling frequency. A 1.28 second hit means 64 samples data for 50Hz and 128 samples data for 100Hz sampling frequency. 128 samples data computational results are better than the 64 and 256 sample data sets. 1.28s is more acceptable than 2.56s drum hit duration like natural drum playing. In this section, we found that 100Hz sensor sampling frequency and 128 sample data set (1.28s) are optimum to check detection.

### C. Acceleration signal noise filtering and axis calibrating

In our experimental sensor, the CPU has 12-bit analogue-to-digital converter (ADC) and an integrated 16-channel. The sensor is configurable over Bluetooth or wired dock connection and has option to configure low-pass, high pass and band filter. We have used the SHIMMER built in function to configure the low-pass filter. A low-pass filter (LPF) passes signals frequency lower than a selected cutoff frequency. To capture sensor data from the accelerometer, the low pass filter shows low noise accelerate than high pass and band filter. To define preferred x, y, z coordinate axis, we have configured 9DoF (degree of freedom) calibration for the Windows platform. The formula below is used to calibrate the tri-axial MEMS 3D accelerometer sensor and write the configuration.

$$c = R \cdot 1 \cdot K \cdot 1 \cdot (u - b)$$

where,  $c = 3 \times 1$  calibrated signal vector,  $R = 3 \times 3$  alignment matrix,  
 $K = 3 \times 3$  sensitivity matrix,  $u = 3 \times 1$  uncalibrated signal vector,  
 $b = 3 \times 1$  offset vector.

### D. Sensor wear position and virtual drum position

The top of the drum stick, wrist, below of elbow and the root of drum stick are the wearable locations of the sensor. The top of the drum stick is the best location of sensor placement for virtual drum accurate detection. The wrist also shows similar accuracy like the top of the stick. But the below of elbow and the root of the stick show low detection accuracy. Due to the comfortable issue and sensor damage security, "wrist" is considered the wearing location. To play the virtual drum, we have selected eight drum position (Figure 4) like a natural drum position. To play the virtual drum, the player needs to guess the drum position and hand posture movement have to move based on position (Figure 3).



Fig. 3. Right side 1- 4 drums hand posture and left side 1-4 same hand posture

In our system, we have indicated a total of eight drum positions, where four positions on the right side and four-position on the left side. We have named the four-position called "first drum position, second drum position, third drum position and fourth drum" for both sensors. The right-hand sensor will cover four positions and the left-hand sensor will cover another four positions. So, we have total two sensors for eight drums and four drum position for each sensor (Figure 4).

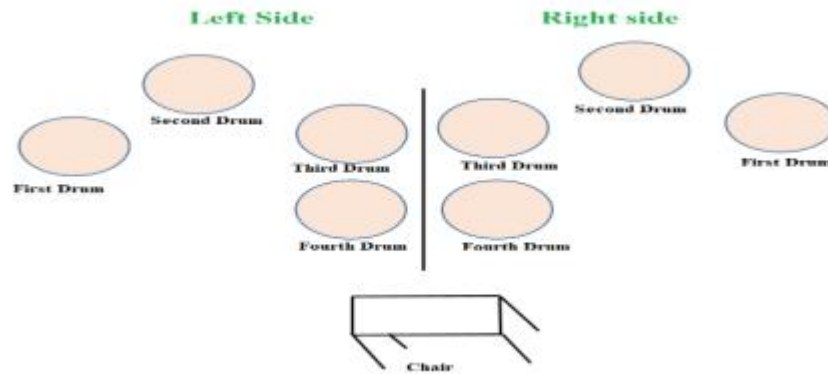


Fig. 4. Position of virtual drum

#### E. Volunteer recruiting and data collection

Volunteers that have good command to play the real drum are recruited for this work. Participants were completely voluntary. The verbal explanation was provided to each volunteer in order to ensure that, participants could understand what our requirements were. Sensors were strapped with the wrist of the volunteer during data record. Via Bluetooth connection and Consensus-basic vendor-provided software, data are recorded in local computer as a CSV file. By plotting the numerical data, we have marked the initial point of each drum activity and separated 128 samples of each drum activity to another sheet as a pure drum activity data set. We have collected 1200 three dimensional accelerometer data set for ten activities (Right side "four drums activities" with "no drum activity" and left side "four drums activities" with "no drum activity"). Here "no drum activity" means any kind of hand posture except "four drums activities" will not be played sound. Though machine learning will detect as one kind of activity but it's has no sound.

#### F. Data analysis and features selection

Based on the data pattern, it is very important to choose the right feature to distinguish activities. Because features are the process of selecting a subset of relevant features for use in detection model construction. It eliminates meaningless and redundant values by keeping significant information. Irrelevant input features create unnecessary computational cost and complexity. For this purpose, we have concentrated more on the data analysis. We applied a lot of statistical and mathematical formulas like mean, standard deviation, variance, proportion, correlation, max, min, range, sum, count, Principal Component Analysis (PCA), Independent Component Analysis (ICA) and fast Fourier transform (FFT) to check which features are able to distinguish drum activities. Among those features, mean, standard deviation and Principal Component Analysis (PCA) show significant threshold value to distinguish drum activities and non-drum activity. Based on data analysis, mean (equation 1), standard deviation (equation 2) and PCA (equation 3) are selected as input features for the machine learning algorithm.

$$\bar{x} = \frac{1}{n} \sum_{k=1}^n x_k \text{ -----1}$$

$$\sigma = \sqrt{\frac{1}{n} \sum_{k=1}^n (x_k - \bar{x})^2} \text{ -----2}$$

$$S = \sum_{k=1}^n (x_k - \bar{x})(x_k - \bar{x})^T \text{ -----3}$$

$$\text{Where } \bar{x} = \frac{1}{n} \sum_{k=1}^n x_k$$



### G. Train and test data preparation

For a high accuracy detection system using a machine learning algorithm, there needs to be an optimum number of training data set from the subject, which will provide good performance. Data from an inexperienced subject make overfitting which leads the machine error detection and reduces overall performance. On the other hand, more subjects than needed do not necessarily lead to improved performance. Sometimes it can affect the performance negatively. Based on volunteer's natural drum playing hand posture, we have selected volunteer and their recorded data. So, we discarded some of the data sets and the remaining 1000 data have been prepared for training and testing (Figure 5). Randomly, the whole data set has been divided into two parts. One (75%) for training data set and another (25%) for testing data set.

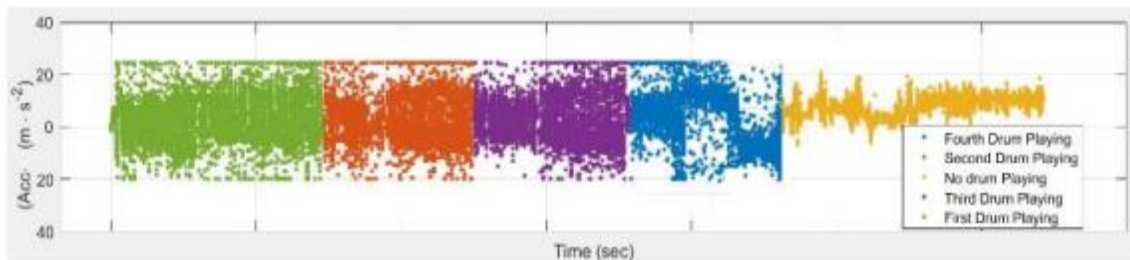


Fig. 5. Virtual drum training and testing dataset

### H. Classifier selection and implementation

To play the virtual drum, need to choose and implement a machine-learning algorithm that will be able to classify virtual drum position and hand posture by recognizing the signal patterns and matching vector of statistical features with the pre-learned ones. We have studied a different type of machine learning algorithm like Neural network, SVM, Apriori, EM, C4.5, complex tree, k-Means, CART, AdaBoost, kNN, Naive Bayes, and PageRank. Based on our work aim and algorithm properties, primarily we have selected two algorithms named support vector machines (SVM) and k-Nearest Neighbors (kNN). These two algorithms have been implemented by using "MATLAB" programming language. In Matlab programming language platform, the mean, the standard deviation and the PCA of training data set are fed into the algorithms. Both classifiers SVM and KNN are trained by the same training data set and the learning rate is found 97.3, 94.2 respectively (Figure 6).

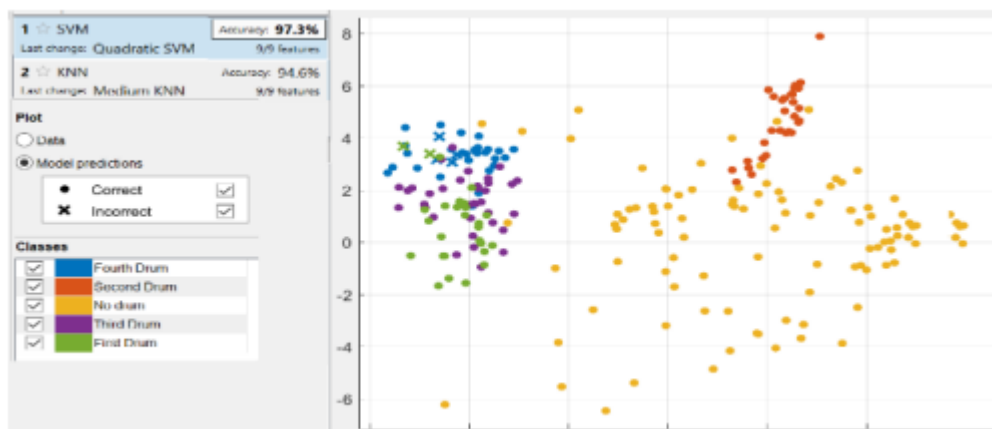


Fig. 6. Classifiers SVM and KNN training and learning rate

Next, we have recorded the virtual drum detection accuracy of each algorithm in simulation environment by using the test data set (Fig 7). Based on the high detection accuracy between SVM and KNN, SVM has been selected (Table II, IV).

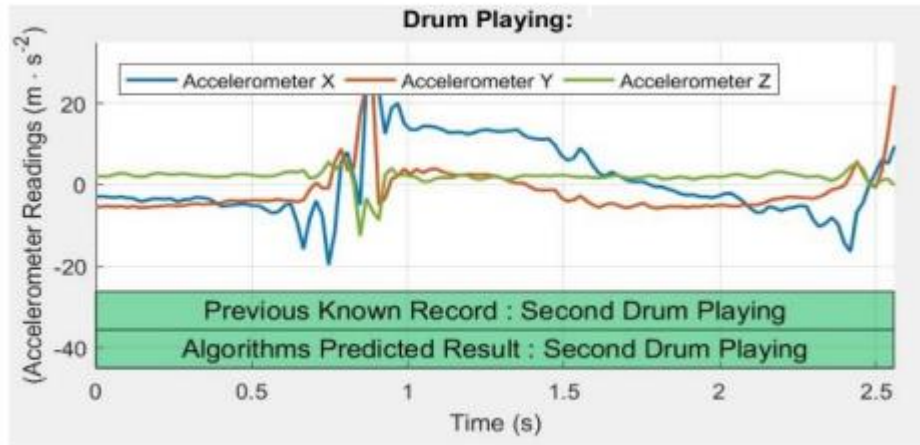


Fig. 7. Testing virtual drum detection at simulation environment

#### I. Real-time hand posture drum detection

MEMS 3D accelerometer provides 100Hz sampling frequency data. This sensor has a built-in MSP430 microprocessor, which has high capacity signal processing and calibrating power. Over the Bluetooth connection, sampling data is stored in the designated computer serial port. We have used open-source software called "Realterm version 2.0.0.70" to scan PC port. Realterm software retrieves the sample data from the designated port and saves in .dat extension file. Using Matlab read-write command, 128 sample data are read from the testdata.dat file. Here "testdata.dat" is our given file name. After extracting statistical features (mean, standard deviation and PCA) of 128 sample data, features are fed to the trained SVM classifier to detect real-time virtual drum detection. To test the real-time detection system, again we recruited volunteers and recorded detection result as well as drum activity video recording to check the real-time detection accuracy (Figure 8).

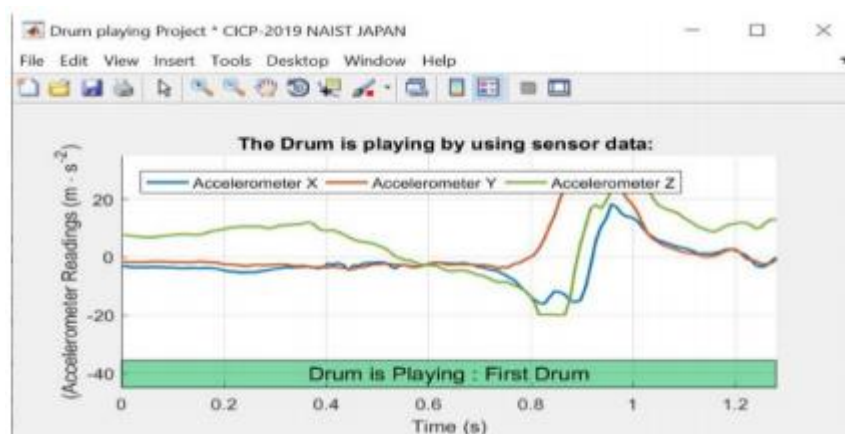


Fig. 8. Testing virtual drum detection at real-time environment

We have compared simulation result and real-time result and found overall 8% low “no detection result” in the real-time environment (figure 12). In the simulation, the test data was collected



from manual observation of the initial point by plotting sample data. But in the realtime system, data's initial point is selected randomly by the system. Due to random initial point selection by the system, it's showing overall 8% no drum detection result. By using windows to overlap troubleshooting, we have solved the problem. Afterwards, we found simulation and real-time similar result. Windows overlap troubleshooting has been discussed in the below section.

#### J. Window size and percentage of overlapping

A window is normally used to segment a data sequence. Due to random initial point selection by the real-time system, the segment of data sequence may not have full data. As a result, the classifier cannot detect the activity. To avoid this problem, we overlapped the window size 50% and observed the result. Although the detection accuracy is as like the simulation result but at the same time, 30-40% same drum two times detect which makes another problem to play virtual drum play. By tuning the window size 10-40%, we have found that 20% is the optimum window overlapping for this work, in which the recorded result is almost similar to our video recording result (Figure 13).

#### K. Drum musical tune selection

In our developed system, there are eight virtual drums which are arranged based on acoustic drum setup as like figure 2 setup. To play the audio the sound, we have collected more than 500 drums audio file from the given "musicradar" website. Primarily, we selected less than a one second audio file for sequential drum playing. Because 128 samples mean 1.28s data storing time. It was sequential processing. Due to non-realistic audio play, we changed our system to parallel processing for realistic audio ( and the details for discussion are in the parallel processing section). Subsequently, we have chosen any play-time audio file based on drum setup and sound pattern.

#### L. Drum Tune and hitting time observation

There are different types of drums that use the acoustic system. Their musical tuning and playing vibration duration are also different. In our virtual system, we have arranged eight drums (right side four and left side four) based on acoustic drum visualization and their names: Ride cymbal, Rack toms, Floor tom, Snare drum, Hi-hat, Crash cymbal, Splash cymbal and China cymbal. By listening manually to the drum music, we have selected the drum sound and included our virtual drum sound system. We had observed the vibration duration of different types of drum by video watching and stopwatch. Based on drum sound pattern and duration of acoustic drum, we have set in our system almost similar audio sound file.

#### M. Multiple hit detection within a fixed interval

By keeping a 100Hz sampling frequency, 128 examples of sample data were needed to detect drum activity. Based on this calculation, one hit detection per second is considered in our develop system. It's a general ideal drum hit duration setup but it may be not always ideal. Sometimes fast-hitting could occur. In that case, we have to detect multiple hits within one window size that means 128 sample data contain double or triple hit. To solve this problem, we applied the peak detection method by calculating the acceleration peak value and window split peak check. Based on the 3D axes value of acceleration, we have recorded the resultant value of acceleration. As a result, by the analysing the peak value of resultant acceleration ( $R = \sqrt{accx.^2 + accy.^2 + accz.^2}$ ), we have recorded the threshold value and included our system to check the peak values. To check double peak, the threshold acceleration value is searched between 1- 60 and 70 -128 interval sample data respectively and 61-69 sample data are kept as a boundary

(Figure 9). If the double peak is found, then the respective drum will play two times the first half of audio again the first half.

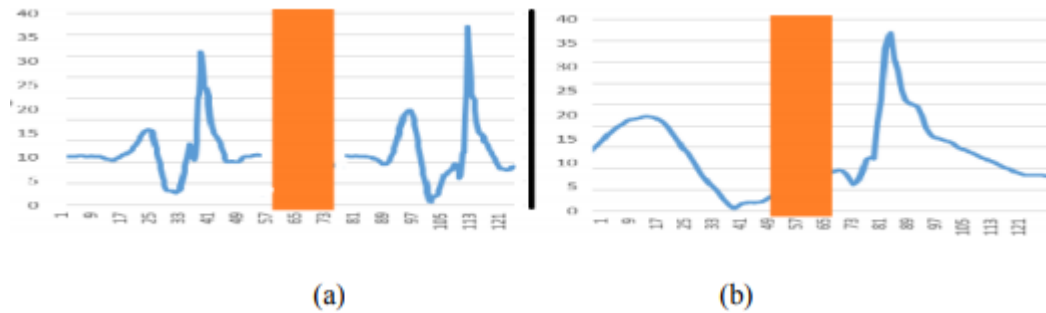


Fig. 9. Checking multiple drum hit (a) Double hit (b) Single hit

#### N. Sound intensity calibration

To play virtual drum, a drum player is needed to apply high force or low force according to song tune. Therefore, it is necessary to change sound intensity with respect to hitting force. Previously, we have calculated the resultant acceleration and added the calibrating equation. From the calibration equation, the intensity level is stored in a temporary variable which affects the sound function to play at a high or low volume according to drum player's force (Figure 10).



Fig. 10. Sound intensity, skeleton architecture and computational system

#### O. Sound tune parallel processing

Latency is a common issue in the detection or recognition system. Because enough data is needed in the buffer to detect accurate detection. Without system-defined data, the detection result has a high probability of mistakes. So more time is needed for buffering enough data. Data reading from the port and calculating the features. Features' feeding to the artificial machine learning algorithm. After processing the input features, the algorithm detects the detection. Next, the peripheral device takes action. So data buffering time, data reading time, features calculating time, algorithm detection time and peripheral device active and action time are needed to count. To perform the step by step processing, latency is inevitable. On the other words, we have to maintain zero-latency for playing the virtual drum; otherwise virtual drum will not produce realistic sound. If any latency to play drum sound means sound - no sound - sound - no sound...which is not acceptable to the listener. The detection system must have latency but our

system has to maintain zero latency, which is a big problem for playing the virtual drum. To avoid this problem, we changed a little bit called "parallel processing". Sound will be played by another thread which has no effect on the consumption of software computation time. The computer's processor will play the sound using the multitasking when our system is commanded. The duration of the audio file for more than one second, based on drum pattern. One sound file is playing, while the other drum files will detect and instantly play. Due to parallel sound processing (Figure 11), the audience will apparently not notice any latency. Although the internal computation detection system has a little bit of latency.

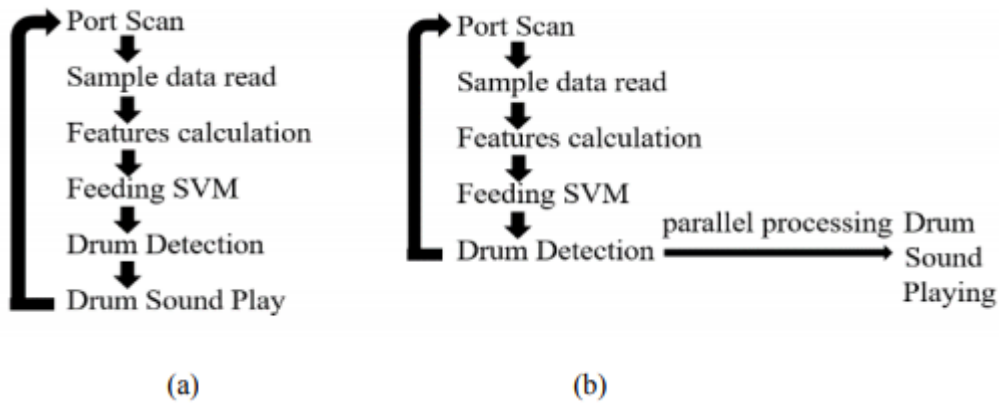


Fig. 11. Sound playing (a) Sequential processing (b). Parallel processing unavoidable

#### 4. RESULT

Support vector machines (SVM) and the k-nearest neighbours (KNN) are primarily used to assess the performance in the simulation environment and the result have been recorded below.

SVM Classifier: Confusion matrix and summary results of virtual drum detection, using SVM classifier are shown in Table I and Table II.

Table I: Confusion matrix of SVM classifier

<b>True Class</b>	<b>Detection</b>					
	<b>First Drum</b>	38		6	1	5
	<b>Second Drum</b>		42		2	6
	<b>Third Drum</b>	4	2	34	3	7
	<b>Fourth Drum</b>	1		7	39	3
	<b>No Drum</b>	1		1	4	44
		<b>First Drum</b>	<b>Second Drum</b>	<b>Third Drum</b>	<b>Fourth Drum</b>	<b>No Drum</b>
<b>Predicted Class</b>						

Table II: Summary results of SVM

Detection	Accuracy (%)	Precision (%)	Recall (%)
First Drum	92.28	76.00	86.36
Second Drum	96.00	84.00	95.45
Third Drum	88.00	68.00	70.83
Fourth Drum	91.60	78.00	79.59
No Drum	89.20	84.00	67.69

KNN Classifier: Confusion matrix and summary result of virtual drum detection, using KNN classifier are shown in Table III and Table IV.

Table III: Confusion matrix of KNN classifier

True Class	<b>Detection</b>					
	<b>First Drum</b>	33	1	9	3	4
	<b>Second Drum</b>		37	1	5	7
	<b>Third Drum</b>	7	3	29	5	6
	<b>Fourth Drum</b>	3	3	7	32	5
	<b>No Drum</b>	2	1	7	4	36
		First Drum	Second Drum	Third Drum	Fourth Drum	No Drum
<b>Predicted Class</b>						

Table IV: Summary results of KNN

Detection	Accuracy (%)	Precision (%)	Recall (%)
First Drum	87.20	66.00	73.33
Second Drum	91.60	74.00	82.22
Third Drum	80.00	58.00	54.71
Fourth Drum	86.20	64.00	65.30
No Drum	85.60	72.00	62.06

By comparing table II and Table IV, we have observed that SVM's accuracy, precision and recall are higher than KNN. As a result, SVM is selected and implemented for real-time drum detection.

#### A. Simulation and real-time detection comparison

After real-time implementation, we have checked the accuracy by video recording and computer-generated result. We found less "no detection" result due to the random data segment selection in real-time (Figure 12).

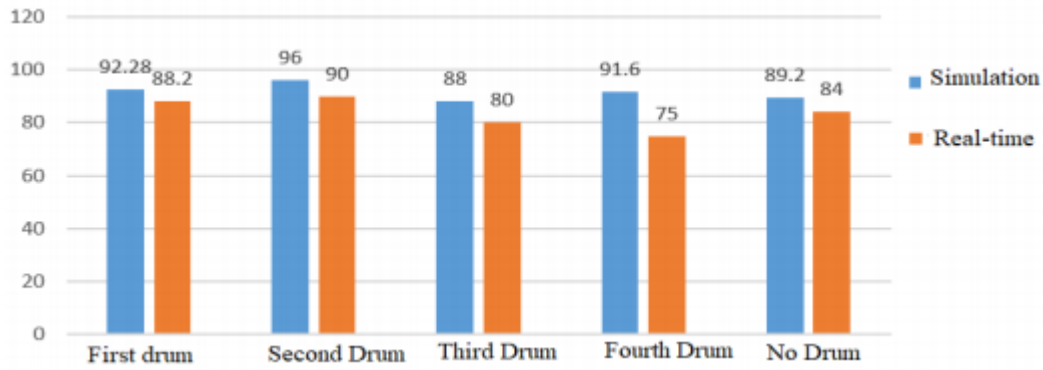


Fig. 12. Simulation and real-time drum detection comparison

### B. Simulation and real-time after 20% windows overlapping

Using 20% windows overlapping, we solved the "no detection" problem and got an almost simulation and real-time (Figure 13) accuracy.

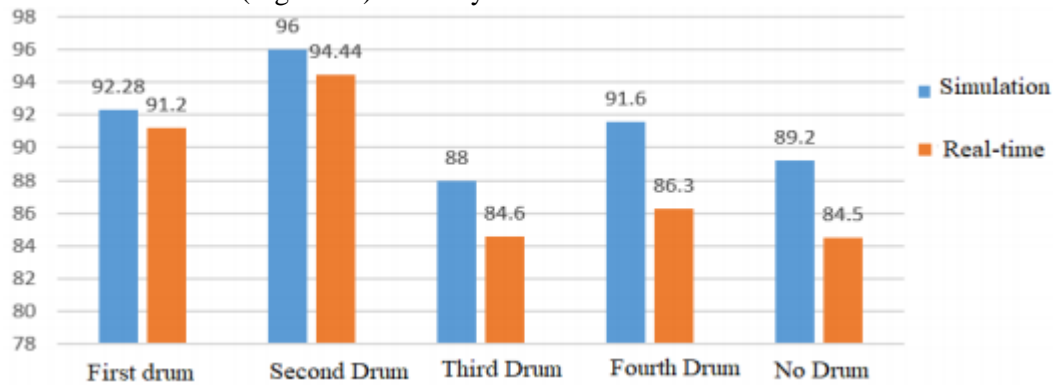


Fig. 13. Simulation and real-time with 20% windows overlapping

## 5. CONCLUSIONS

In this work, we present a difference application of data as an entertainment in music field. Step by step, we have overcome the arrival problem by quantitative data analysis and optimum decision. In this "Playing Virtual Musical Drums", decisional phase is Sensor type - accelerometer; dimension - 3; brand -SHIMMER; sampling frequency - 100Hz; windows size - 1.28s; sample number 128; signal noise filtering-low pass; sensor wearing position - wrist; hand posture - 4; virtual drum position - 8; skilled volunteer recruiting; features - mean, standard deviation and PCA; Classifier - SVM; windows overlapping - 20 are selected. We have performed a simulation in real-time by training the data, testing the data and video recording. We have observed drum hits over time, drum musical patterns and the playing time. We have implemented multiple hit detection, sound calibration and sound parallel processing. After completing the steps with 88.20% detection accuracy, we have played the virtual drum successfully like an acoustic drum. In Figure 14, virtual drum playing have shown by the user.



Fig. 14. Playing virtual drum by using MEMS 3D accelerometer sensors

## 6. FUTURE WORK

In our virtual drum has some latency. By increasing sensor frequency, latency will be removed in future and virtual drum detection accuracy will also be improved.

## CONFLICTS OF INTEREST

All authors declare no conflicts of interest.

## ACKNOWLEDGEMENT

This work was supported by Creative and International Competitiveness Project (CICP)-2019 of NAIST and partially supported by the Ministry of Education, Culture, Sports, Science, and Technology of Japan (16K07223 and 17K00406) and NAIST Big Data Project.

## REFERENCES

- [1] Joseph J. LaViola Jr., "3D Gestural Interaction: The State of the Field," *ISRN Artificial Intelligence*, vol. 2013, Article ID 514641, 18 pages, 2013.
- [2] Berman, Sigal, and Helman Stern. "Sensors for gesture recognition systems." *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42.3 (2011): 277-290.
- [3] Chen, Ke-Yu, Shwetak N. Patel, and Sean Keller. "Finexus: Tracking precise motions of multiple fingertips using magnetic sensing." *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 2016.
- [4] Truong, Hoang, et al. "Capacitive sensing 3d-printed wristband for enriched hand gesture recognition." *Proceedings of the 2017 Workshop on Wearable Systems and Applications*. ACM, 2017.
- [5] MEng, Kang Li. "Development of finger-motion capturing device based on optical linear encoder." *Journal of rehabilitation research and development* 48.1 (2011): 69.
- [6] Brown, Dom, et al. "Leimu: Gloveless music interaction using a wrist mounted leap motion." (2016).
- [7] Nymoen, Kristian, Mari Romarheim Haugen, and Alexander Refsum Jensenius. "Mumyo—evaluating and exploring the myo armband for musical interaction." (2015).



- [8] Serafin, Stefania, et al. "Virtual reality musical instruments: State of the art, design principles, and future directions." *Computer Music Journal* 40.3 (2016): 22-40.
- [9] Caramiaux, Baptiste, and Atau Tanaka. "Machine Learning of Musical Gestures." NIME. 2013.
- [10] Torre, Giuseppe. "The design of a new musical glove: a live performance approach." (2013).
- [11] Farhad et al. , "A Direction-Sensitive Fall Detection System Using Single 3D Accelerometer and Learning Classifier", IEEE International Conference, MediTec, Dec 2016.
- [12] Farhad et al., "Real Time Direction-Sensitive Fall Detection System Using Accelerometer and Learning Classifier", IEEE 4th International Conference ICAEE -2017
- [13] <http://www.shimmersensing.com/about/> [accessed last on . Dec 18, 2019]

## AUTHORS

**Shaikh Farhad Hossain** received his M.Sc. degree in Information and Communication Technology from Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh, in 2017. He is currently pursuing his PhD in Biomarkers, Proteins and Gene expression in Nara Institute of Science and Technology (NAIST), Nara, Japan. He received the best paper award in the IEEE 4th International Conference on Advances in Electrical Engineering (ICAEE), September, 2017, Dhaka, Bangladesh.



**Shigehiko Kanaya** received B.Sc. degree in Bio- science from Science University of Tokyo, Japan in 1985, and Ph.D. from Toyohashi University of Technology, Japan in 1990. He served as an Assistant Professor in Information Engineering at Yamagata Univ. in 1990, Guest Associate Professor at National Institute of Genetics in 1996, Associate Professor at Electronic and Information Engineering in 1999, Associate Professor, Applied Bio system Engineering at Yamagata Univ. in 2000, Guest researcher at Bio radical Institute (Yamagata Prefecture), in 2000 Associate Professor, Research and Education Center for Genetic Information at NAIST in 2001, Associate Professor, Graduate School of Information Science at NAIST in 2002, Professor, Graduate School of Information Science at NAIST in 2004 and is currently working as a Professor, Nara Institute of Science and Technology, Japan.



**Md. Altaf-Ul-Amin** received B.Sc. degree in Electrical and Electronic Engineering from Bangladesh University of Engineering and Technology (BUET), Dhaka, M.Sc. degree in Electrical, Electronic and Systems Engineering from University Keban gsaan Malaysia (UKM) and PhD degree from Nara Institute of Science and Technology (NAIST), Japan. He received the best student paper award in the IEEE 10th Asian Test Symposium. Also, he received other best paper awards as a co-author of journal and conference articles. He previously worked in several universities in Bangladesh, Malaysia and Japan. Currently he is working as an associate professor in Computational Systems Biology Lab of NAIST. He is conducting research on Network Biology, Systems Biology, Cheminformatics and Biological Databases. He published around 70 peer reviewed papers in international journals and conference proceedings. Current google scholar citation index of his publications is more than 4600.

