Hawkins G., Chew E. (2022) TOM: The Assistant Robotic Tutor of Musicianship with Sound Peak Beat Detection. In: Ab. Nasir A.F., Ibrahim A.N., Ishak I., Mat Yahya N., Zakaria M.A., P. P. Abdul Majeed A. (eds) Recent Trends in Mechatronics Towards Industry 4.0. Lecture Notes in Electrical Engineering, vol 730. Springer, Singapore. https://doi.org/10.1007/978-981-33-4597-3_67

TOM: the Assistant Robotic Tutor Of Musicianship with Sound Peak Beat Detection

Gareth Hawkins[1] and Esyin Chew[2 [0000-0003-2644-9888]]

[1, 2] EUREKA Robotics Lab, Cardiff School of Technologies
Cardiff Metropolitan University, Western Avenue, Cardiff CF5 2YB, UK.
[1] G.Hawkins@outlook.cardiffmet.ac.uk
[2] echew@cardiffmet.ac.uk

Abstract. Most past literatures surrounding musical robots has focused on either the engineering design of physical robots that could produce musical sounds, or on the mixed mechatronics aspects of algorithms enhancement to design robots gaining musicianship skills. The recent combined both these research fields into one is called Robotic Musicianship, which is innovative but with limited literatures. Robot musician involved sound peak beat detection techniques to enable musical perception. Responding to the novelty demand, the study adapted design approach of onset beat detection, blended with positivists' philosophical approaches, and deductive principles associated with these philosophies were adhered to. The study aim to conduct a conceptual design and implement the drum rudiment for a robotic assistant tutor to co-demo electronic snare drum. The design approach of onset beat detection is adapted. The fundamentals of this robot prototype and rudiment work well, although many improvements can be done on the functionality and fluidity, doing so should further enhance engagement and musical development, and achieve the long-term goal to branch out to higher levels of education for music.

Keywords: Robotics, Robot application, Robotic Assistant Tutor, Robotic Musician, Sound Peak Beat Detection

1 Motivation and Bibliometric Investigation of Robotic Musicianship

Due to the new norm of the COVID-19 social distancing and the lacking of resources of accessing drum tutor, this research investigates the past literatures of robotic musician, followed by the preliminary design of an assistant robotic musician as a tutor for beginner drummers. There are two primary research areas within Robotic Musicianship: Musical Mechatronics and Machine Musicianship. Musical mechatronics is focused on constructing physical robots that are able to produce musical sounds with motors, solenoids and gears, come in many shapes and sizes and have been developed to learn many different musical instruments over the years [1]. For examples: (1) StrumBot a six-stringed robotic electric guitar that uses a dual-pick strumming arm in order to pluck guitar strings [2]; (2) Cog is another upper-torso humanoid robot with

21 degrees of freedom, and has various sensory systems, including visual, auditory and tactile senses [3]. Cog was built with compliant arms (each with 6 DoF), instead of using stiff arms, as it was more suitable for a robot that would interact with an unknown environment. These compliant arms were controlled by non-linear oscillators, which provided energy to generate motion [4]. Auditory feedback was used to modify the oscillator behaviour, which moved the arms. After receiving raw audio, it was processed and thresholded to find the sound of the drum. An auditory-motor loop was developed that combined auditory feedback and auditory processing, which made the robot listen to its own drum hits, thus making the robot continuously hit the drums.

Machine Musicianship is focused on developing algorithmic expression, music cognition [5] and its foundation is derived from the fields of music theory, which involves analysing, performing and composing music with computers. Implementing such theoretical models usually ranges from simple musical problems to live interactive musical compositions. Some systems have been developed that employ machine musicianship. Systems such as EMMY and Emily Howell [6-7] was programmed to find a set of instructions or creation of a musical piece with note-to-note analysis, then creates scale degrees that are similar to the original style it learnt from, in order to produce new styles instead of replicating music style, and could further communicate with users in different languages, to expand collaboration with different cultures [6]. HAILE is the robot percussionist capable of perceptual, physical and socials aspects of musicianship, to analyse musical input from a human, in real-time [7]. HAILE was programmed with improvisation algorithms, which enabled the robot to detect multiple musical qualities, such as pitch, volume and rhythmic elements. To encourage human-robot interaction, the algorithm transforms and modifies human input to create a similar improvised output which a human player would react to, during performances. MahaDeviBot was the combination of ideas taken from music robotics, hyperinstruments and machine musicianship that uses a symbolic Music Information Retrieval (MIR) based approach [1]. The MIR approach enabled the robot to automatically supply its database with drum beats, learnt from pre-programed human input. How the robot performed the drum beats autonomously was from accessing its database of drum beats, using sensor-based retrieval that included a set of queries received from a human sitar player, in real-time. Shimon is an autonomous, improvising Marimba player [8], a change from the usual percussive instruments and is the successor to HAILE. The approach used for Shimon, is different compared to previous works, it uses a gesture-based behaviour system that focuses on physical movement and anticipates said movements for input, instead of the conventional auditory approach [8]. Shimon uses a non-humanoid head for social communication that provides organic movement, and is equipped with a camera-like shutter in order to convey emotion.

The two common approaches used for beat detection are autocorrelation analysis and onset component detection, two robots previously mentioned, HAILE [7] and MahaDeviBot [1] both utilise onset beat detection. Nevertheless, there has been newer approaches within beat detection for robotics, using a NAO robot by combining the two common methods of beat detection which involves estimating a global tempo by analysing autocorrelation of the onset features from musical pieces [9, 10]. A newly proposed method for beat detection was developed for a dancing robot, they proposed a multi-model method that combines both auditory and visual tracking [11]. Their beat tracking method is based upon the work from Murata, et al. [12], that involves calculating the autocorrelation of onset features, similar to Xia et al.'s approach [9]. What makes this approach distinctive, is that the inclusion of a visual tracking system which uses a Kinect camera that tracks the skeleton of a dancer and their movement of joints. Both audio and visual tracking systems work together to estimate the tempo and beat time from the music and human dancing. From their experimental results, it showed that their proposed method always outperformed beat-tracking methods, and on average it over performed other related methods too. Although, their results did show that the visual tempo estimation failed on occasions that included small movement with the hands and feet. This could suggest that this method might not be suitable for visual tracking for body movements from musicians that perform small movements, for example movements such as quick alternate guitar strumming, or soft drum hits that require small movements.

2 Research Method and Design

We stand a positivists' philosophical approaches with deductive principles, to conduct a conceptual design and implement the drum rudiment for a robotic assistant tutor to co-demo electronic snare drum. In this study, the conceptual design is grounded on the design approach of onset beat detection that has been observed in Weinberg's, Hoffman's and Kapur's research [1, 7, 8], which shows heavy usage of auditory features within robotics utilising beat detection methods. Figure 1 shows the initial conceptual design for the beat detection approach used in this study, and onset beat detection serves as a good departing point for Robotic Musicianship research and development. Other NAO robots have used this approach or a higher level of this approach, as seen from Xia et al. [9], however, this design would be for a different scenario, that being a drum tutor assistant. Other methods were considered such as, the multi-modal method from Ohkita et al. [11] to use for research.

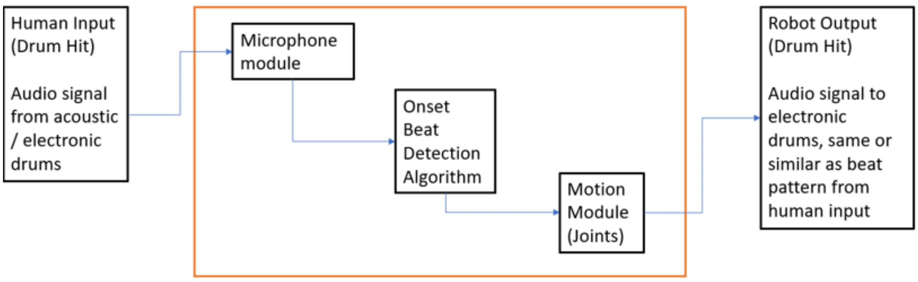

Fig. 1 Conceptual Beat Detection Design for Drum Tutor Assistive Robot

The overall research approach is focused more on machine musicianship design rather than music mechatronics, but understanding the methods used within musical mechatronics has provided useful insights, such as the visual features used for tracking of humans [11], instead that could potentially be used to track the drums in the exploration. Computation and semi-autonomous responses to beat and music will be the main focus, while some psychology could be involved. Since the NAO robot is already built, although construction of the extended 3D printed accessories could be present, if the robot needs it to support the musical process. While robots have been introduced as teaching assistants, and the same with robots being involved with musical tasks, none have been combined both together and seen how introducing a robot into one on one music teaching would affect the learning experience, in a drum lesson scenario.

3 Robot Tom Design and Implementation

3.1 The Design of an Assistant Robotic Tutor Of Musicianship

A drum lesson scenario is going to be used to help design the robot drum teacher assistant. In the drum lesson scenario, there will be 3 actors, a human teacher, a human student and the robot teacher assistant, TOM. The teacher will begin the drum lesson by powering on TOM. When TOM is powered on, the teacher can control TOM's hands by talking to it. Speech commands such as "open" and "close" will tighten or loosen the grip of the hands (see Fig 2), this enables the teacher to place drumsticks into TOM's hands. Once the drumsticks are in position, the teacher can proceed onto the next step, which is activating the three different mode that enables TOM to play drums.

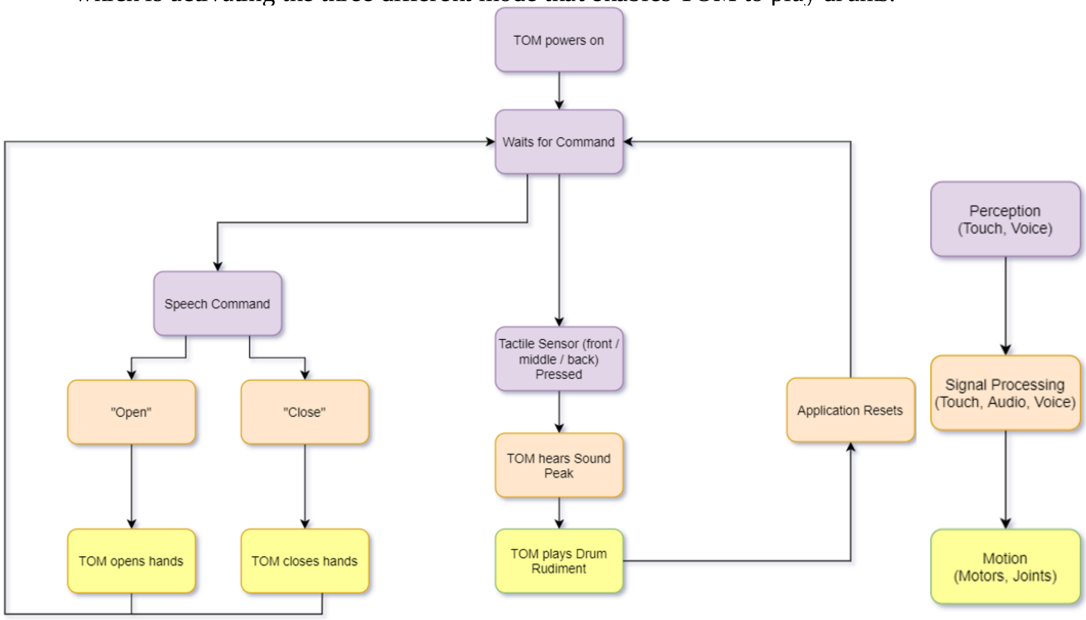

Fig. 2 Drum Lesson Scenario, Robot Application Overview

The three modes are activated by touching the tactile head sensors on TOM, which will start the sound peak detection module, ready to receive auditory input. The student will provide an auditory input by playing one crochet note on the drum snare, TOM will listen for the peak sound from the student's crochet note, and will then proceed to play a drum rudiment. The sound peak detection will continue to run as this provides the teacher and student to continuously play and practice the drum rudiment. Although, it is important to note, that before moving onto a different mode, the current mode must first be reset, otherwise the sound peak detection module from the previous mode will still be active and will pick up auditory input, thus confusing TOM on which drum rudiment to play. After TOM has finished playing the drum rudiment, the student is expected to replicate the drum rudiment. When the student is playing the drum rudiment, the teacher can then observe and evaluate the students playing (see Fig. 2).

At current development, the robot only plays simple drum rudiments, this is because drum rudiments are one of the fundamentals of drumming, and is essential to playing drums. This replicates the process of how traditional drum teaching is currently taught and relates to the way a human would learn drums, by starting with the basics. This also provides a small and suitable scope for a robot drum teacher prototype, that can be tested to students who are beginners or completely new to playing drums. The design of this program is not intended to replace a music teacher but rather aid them in their work. The design is meant to encourage human-robot collaboration and interaction for both human actors (teacher and student) and provide an interesting experience for students to help them learn and give them motivation to continue learning. Fig 3 presents the different types of interaction that is used within this application:

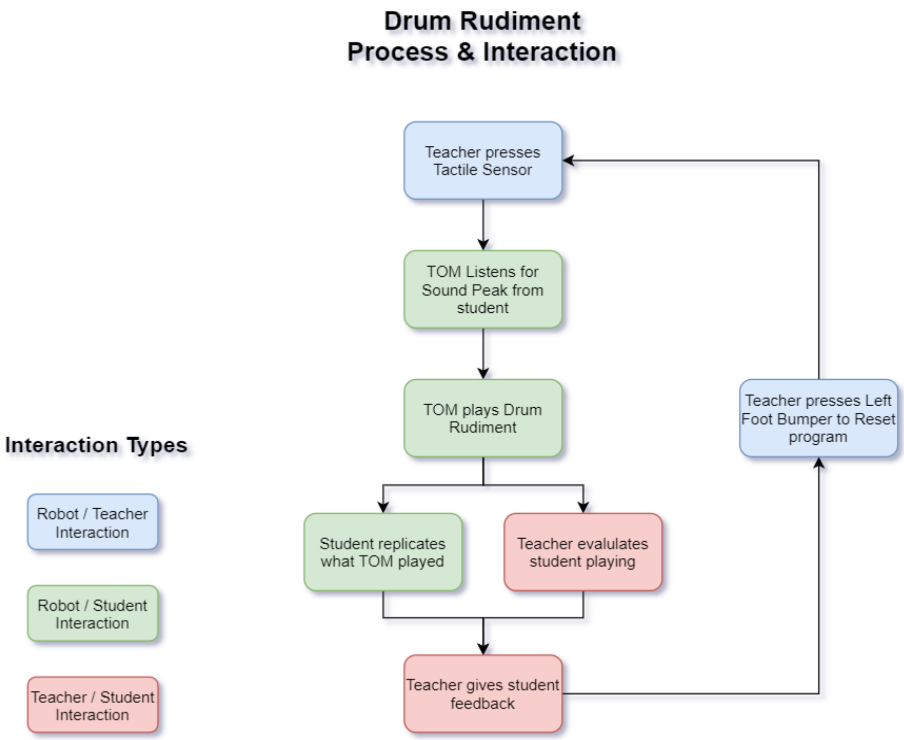

Fig. 3 Drum Lesson Scenario, Robot Application Overview

3.2 The Implementation of Assistant Robotic Tutor Of Musicianship

IDE, Speech Recognition and Tactile Head Sensors - Three Drum Rudiments

In order to implement the above designs onto a NAO robot, the Integrated Development Environment (IDE), Choregraphe 2.8 was used, which is built for developing applications for NAO. To help program NAO, Choregraphe's built-in framework, NAOqi framework, was used to call many core modules (touch, audio, motion), that can communicate with each other. Alongside using the NAOqi framework to help program the application, the Python Software Development Kit (SDK) was used. To implement Speech Recognition into TOM, the Speech Recognition box was used, which enables TOM to recognise a word from a list of words, e.g. 'open' and 'close'. The outputs on the Switch case box are then connected to custom Python script boxes to control both the hands on TOM, correspondingly labelled 'Open Hands' and 'Close Hands', by using the 'angleInterpolation' method. The parameters for angleInterpolation are as followed: (jointName, targetAngles, targetTimes, isAbsolute). Therefore, the first motionProxy sets the initial position of the hands to 0%, which makes both hands close. The second motionProxy changes the position of the hands to 30% within 0.5 seconds (see Fig 4), which opens up or close the hands in order to loosely place a drumstick in the hands. Speech recognition could also be used to help implement new features such as changing the robot's position for different drum grip techniques, or for hitting different drums on a drum kit.

The Tactile Head box, which detects touch for all tactile head sensors, was used to implement the three tactile head sensors functions and make them perform three different drum rudiments. The tactile head box features If statements which states that, if the signal value is greater than 0, transmit a new signal to outputs of the tactile head box. The front tactile head sensor is used to play the first drum rudiment which enables the robot to play 1 crotchet note on the drums, using the right arm. The middle tactile head sensor is used to play the second drum rudiment, which again uses the right arm, but instead plays 4 crotchet notes on the drums. Finally, the rear tactile head sensor is used for the third rudiment, which enables the robot to play 4 crotchet notes on the drums, with both arms alternating with each crotchet note played. To activate each tactile head sensor, a user must make a tactile gesture. The signals generated from the tactile gestures are sent to the ALTouch module, by using the ALMemory module which triggers the memory events, FrontTactilTouched, MiddleTactilTouched and RearTactilTouched, which are present within the Tactile head box.

Sound Peak Detection

Being able to process human touch to signal the tactile head box, TOM requires a mean to play these rudiments through sound detection methods. This was achieved by using the Sound Peak box, which outputs a signal whenever a sound peak is detected. The Sound Peak box utilises ALSoundDetection, a NAOqi audio module. ALSoundDetection detects and processes significant sounds, such as a drum hit, from the front facing microphone at 16000Hz, in OGG format. In conjunction with using this audio module, the core module ALMemory is used, and the memory event 'SoundDetected', is called upon when a significant sound have been detected. Both these modules communicate with each other to process audio signals. The way the sound processing works, is very similar to the Root-Mean-Squared (RMS) method of calculating an average audio signal overtime. When a signal is detected and passed through to the ALSoundDetection module, the raw signal is levelled by calculating the average signal, then the peak sound from this averaged signal is captured (see Fig. 4). An array of sound elements is then generated, which is stored in an 'index', that contains all the indexes of the starting and ending points of the peak sound detected. To use this index, it must be sent through by calling upon the SoundDetected memory event, in the ALMemory module. The SoundDetected event organises sound elements as followed:

[[index_1, type_1, confidence_1, time_1], ..., [index_n, type_n, confidence_n, time_n]] (Doc.aldebaran.com, 2020c).

Where "n" represents the number of sound elements found in the peak sound detected. Each index will have a "type", where type_1 represents the start of the sound, and type_0 represents the end of the sound captured. So, to retrieve the peak sound from this SoundDetected event, each element must be analysed and when type_1 is found, a signal is stimulated. This is achieved from the Array Streaming script that is present in the Sound Peak box. This script selects each element one at a time, and if the sound index equals to 1, then a signal is stimulated to the next box function. The reason for using this RMS method to create an average signal, is that it is easier for TOM to perceive, than a raw audio signal. When analysing raw audio waveforms, a random peak sound could occur when recording and TOM could miss that peak sound and not stimulate any signals. But, by calculating the RMS power level of the continuous waveform, that was created by the drum hit, it makes it easier for TOM to perceive and analyse the audio signal, which makes this function more reliable with sending power signals to other functions.

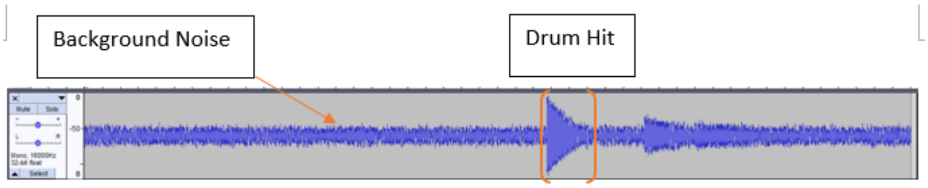

Fig. 4 Drum Hit Audio Waveform Recorded from NAO's Microphone, Visualised in Audacity

The output signals that the Sound Peak boxes produce are sent to Timeline boxes. These timeline boxes are used to create custom animations that TOM performs, in order to hit the drums. There are three timeline boxes used in this application, each of which represent the three previously talked about drum rudiments that will be played in the drum lesson.

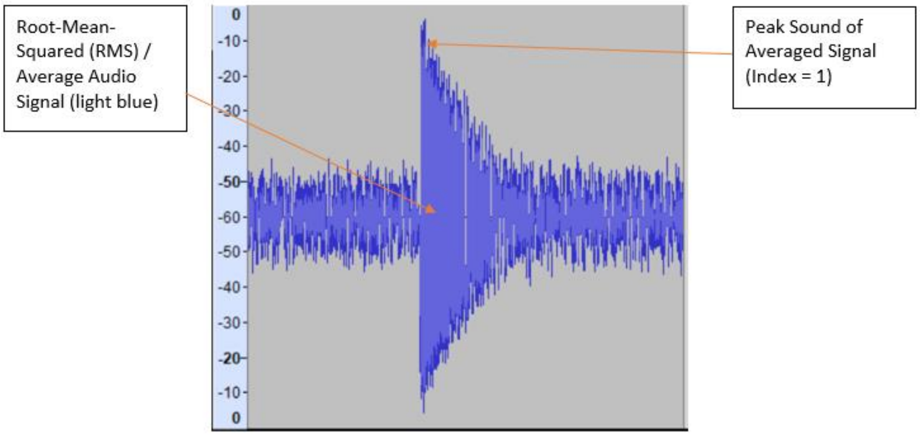

Fig. 5 Close up of Drum Hit Only Waveform

Animation and Reset – Arms and Bumpers

The custom animation process focuses on the movement of the arm joints. This involved using the Inner Motion Timeline, Robot View, the Inspector and a Physical Robot. The physical robot was essential to this process, as custom animations cannot be made with a virtual robot in Choregraphe 2.8. When a keyframe is selected in the Inner Motion Timeline, it saves the current position of the robot into that keyframe. The robot view was used to choose an arm joint that wanted to be moved, once chosen the inspector would appear with multiple actuators associated with the arm joint. In the inspector, the actuators were adjusted by using both the sliders and entering degree values to move the actuators into the desired position. Through vigorous testing of multiple actuator positions and keyframe placements, suitable animations and motions were designed and developed for each drum rudiment.

By discovering the complexity of creating custom animations with python, comparing it with the previous custom python scripts, and how the Timeline box provide quick and easy to create animations, it made sense to continue using the Timeline boxes to develop the rest of the animations for the other drum rudiments, to help alleviate the time constraints for this project. The bumper sensors located on both feet of TOM were used to implement the Reset function that was needed for this exploration. Once the sound peak boxes are activated, they will continue to run even when moving onto the next drum rudiment. This makes multiple sound peak boxes listen for sound peaks at the same time, which will confuse TOM on which drum rudiment to play, and result with potentially playing the wrong drum rudiment.

The positive and negative feedback was taken into consideration on what features should be focused on, and how the strengths and weaknesses can be addressed in future works.

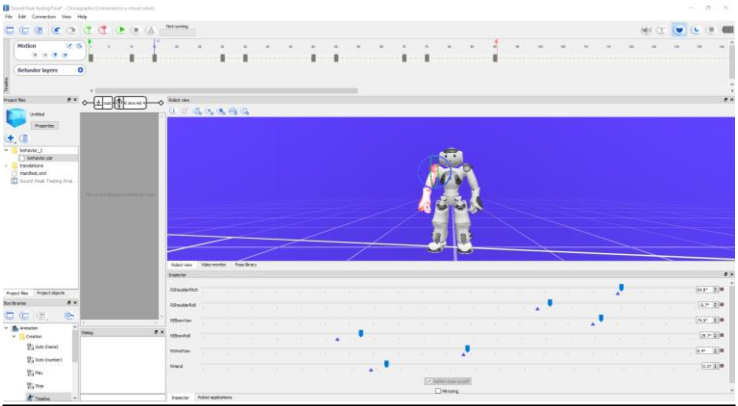

Fig. 6 Inner Motion Timeline, Robot View, Joint Inspector

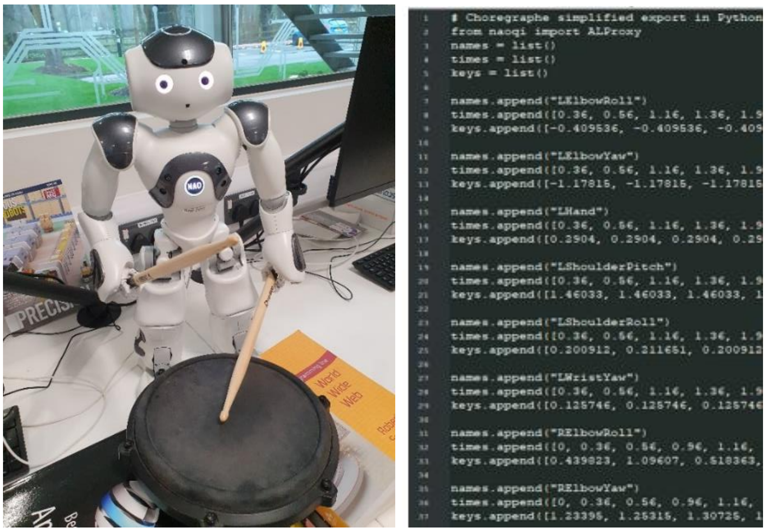

Fig. 7 Robot Tom played Electronic Snare Drum Testing & Exported Animations to Python Code, from first drum rudiment Testing

4 Reflection and Conclusion

An investigation has been conducted on how a prototype Robotic Musicianship can be used as a learning tool, which enhances engagement and musical development, in a drum lesson scenario. TOM is capable of beat detection by using basic sound peak detection, on drum hits that a student or teacher performs. The tactile sensors provides different modes that accommodate the multiple drum rudiments that TOM can play [13]. The appearance and basic movements makes TOM suitable and engaging for teaching younger audiences and would be appropriate for lower levels of drum teaching. Additional drum rudiments could also be implemented in the future by using new and different tactile gestures. However, while TOM can perform basic drumming techniques, it is however limited or not suitable for advanced drum teaching that requires better functionality and fluidity for advanced drumming techniques. The move and force from TOM, is also limited. This presents less variation in drum hit velocity, which limits the ability of performing different drum styles, and dynamics. The sound detection functionality can be enhanced further, with increasing the accuracy in detection for background noise removal, e.g. only capturing instrumental audio and not conversations from humans. At a concert setting, the issue of unwanted mechanical sounds would not matter much, but within a classroom setting, the noise of the robot could cause distractions. While replacing components worked for Strumbot [2], it could cause more problems than solutions when using a NAO robot, but considering this issue and solution would be useful for the future when constructing an original robot for a long term research and design in the next phase.

References

1. Kapur, A., A History Of Robotic Musical Instruments. ICMC, 10.1.1.88.4599 (2005).
2. Vindriis, R., Carnegie, D., StrumBot-An Overview of a Strumming Guitar Robot, Proceedings of the International Conference on New Interfaces for Musical Expression, Queensland Conservatorium Griffith University, pp. 146–151. NIME, pp.146-151 (2016).
3. Brooks, R., Breazeal, C., Marjanovic, M., Scassellati, B., Williamson, M., The Cog Project: Building a Humanoid Robot. Lecture Notes in Artificial Intelligence, 1562, DOI: 10.1007/3-540-48834-0_5, (2002).
4. Brooks R.A., Breazeal C., Marjanović M., Scassellati B., Williamson M.M., The Cog Project: Building a Humanoid Robot. In: Nehaniv C.L. (eds) Computation for Metaphors, Analogy, and Agents. CMAA 1998. Lecture Notes in Computer Science, vol 1562. Springer, Berlin, Heidelberg (1999).
5. Rowe, R., Machine Musicianship. MIT Press, Cambridge, MA, USA (2004).
6. Cope, D., The well-programmed clavier. XRDS: Crossroads, The ACM Magazine for Students. Association for Computing Machinery (ACM), 19(4), pp. 16. (2013)
7. Weinberg, G., Driscoll, S., Parry, R.M., Haile - An Interactive Robotic Percussionist, HRI'07, March 8-11, Arlington, Virginia, USA (2007)
8. Hoffman, G., Weinberg, G., Shimon: An interactive improvisational robotic marimba player. Conference on Human Factors in Computing Systems - Proceedings, pp. 3097–3102. DOI: 10.1145/1753846.1753925, (2010).
9. Xia, G., Tay, J., Dannenberg, R., Veloso, M., Autonomous Robot Dancing Driven by Beats and Emotions of Music. AAMAS '12: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1, pp. 205-212, (2012)
10. Nao, Softbank. www.softbankrobotics.com/emea/en/nao, last accessed 2020/06/18.
11. Ohkita, M., Bando, Y., Ikemiya, Y., Itoyama, K., Yoshii, K., Audio-visual beat tracking based on a state-space model for a music robot dancing with humans. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 5555-5560, 28 - Oct 2, 2015. Hamburg, Germany (2015).
12. Murata, K., Nakadai, K., Takeda, R., Okuno, H. G., Torii, T., Hasegawa, Y., Tsujino, H. A beat-tracking robot for human-robot interaction and its evaluation. Humanoids 2008 - 8th IEEE-RAS International Conference on Humanoid Robots, pp. 79-84, Daejeon (2008).
13. Robot Tom, YouTube Demo (2020) www.youtube.com/watch?v=56Nl4tGbjAA