I Am An Instrument

Human-Machine Interaction Using A Gesture-Controlled Generative Music Feedback System

Benjamin Selig

179034778

Supervisor: Dr. J. Graham-Harper-Cater

Assessor: Dr. N. Johnston

Integrated Mechanical and Electrical Engineering
MEng Final Project Report
2022

Department of Electrical and Electronic Engineering
University of Bath

9501 words

Acknowledgements

I would like to thank my mother, Naomi, partner, Cecilia, and sister, Leah, who have been immensely supportive and encouraging throughout my degree. I would also like to thank my supervisor, Johnathan Graham-Harper-Cater for his guidance throughout this project. Finally, I would like to thank all the participants who took part in the user study.

Abstract

Audio feedback for rehabilitation or public use is an increasingly popular research field. The ability to transform an individual's personal efficacy [4] or understanding of public spaces [29] is greatly enhanced by the creation of virtual interactive environments. Developments in computer vision have enabled human-machine interaction in unrestricted operating spaces, thereby conveying information with far greater intention. This report aims to provide insight into the challenges and opportunities that come with designing an low-skill interactive generative audio feedback system when using multiple gestural control methods. A generative audio system consisting of a granular synthesis module was parameterised to user-friendly control parameters to enable the mapping of incoming gesture data to control the pitch, tempo and gain of three instruments[1]. Results indicated that the use of hand position was the most intuitive control mechanism, however, activating instruments using body gestures stimulated greater movement and a "full body experience". This suggests that this second mechanism could be better suited to use in public or clinical contexts. All users preferred the ability to continuously control parameters compared to only having discrete parameter control. The use of generative music was effective and engaging, with 89% of users thoroughly enjoying the sound generated. The use of generative music in interactive audio feedback for clinical applications may be more appropriate than the use of pre-composed music.

[1] https://github.bath.ac.uk/bs704/FYP-I-Am-An-Instrument

Table of contents

List of figures

List of tables

Chapter 1

Introduction

The use of artificial intelligence to re-conceptualise traditional ideas surrounding music and sound has become an increasingly relevant field of research. Advancements in Generative Adversarial Networks and Deep Learning, such as WaveNet [36], WaveGAN [12] and GANSynth [14], have enabled systems to learn from unstructured raw audio data to generate music that has a realistic sound and texture. Machine learning can be utilised as a tool for the design of Digital Musical Instruments that make musical decisions in response to interpreting sensory data from their surrounding environment. This enables interactions that are far more sophisticated than those designed using conventional mapping strategies. Such systems can employ "human-in-the-loop" methods that use human interaction as input to an audio feedback system [45, 51]. The use of audio feedback has also been explored in medical and sports contexts to encourage movements for real-time performance and spatial awareness [42, 20]. There is, however, no knowledge of research that has evaluated the different ways humans interact with such systems in the context of music creation. An improved understanding of the ways in which people interact with audio when using movement offers the potential for performers and the public to learn from and engage with sound in new and meaningful ways.

1.1 Context

The operating space of traditional human-machine interaction techniques, such as a mouse, keyboards and buttons, are typically fixed and therefore are unable to meaningfully convey the user's intention. Developments of human-machine interfaces have allowed humans to move in unrestricted operating spaces thereby conveying information with far greater intention and efficiency [24]. These interactions have been

facilitated by advancements in both wearable technologies such as data gloves, and also contact-free interaction that employ computer vision. From stroke rehabilitation to music composition, the applications of such technologies are extensive.

Rehabilitation plays a crucial role in maximising recovery for both neurological and orthopaedic patients. Traditional rehabilitation methods, based on one-to-one therapist-patient interactions, and which use external devices to strengthen specific target areas are both difficult and time-consuming, particularly for patients recovering from severe strokes [50]. The use of audio-visual feedback in virtual and augmented reality rehabilitation systems has been an interesting research field because of the ability to transform the user's self-efficacy (i.e., the user's belief in their capacity to execute a certain behaviour) produced by a sense of greater emotional engagement and perceptual stimuli [38]. These systems serve to stimulate greater active limb motion through the full range of motion, making them superior to interventions that only use passive motion (i.e. movement through use of an external force) [4].

Similarly, virtual environments have facilitated unique experiences for the public to engage in culture and their surrounding environment. Audio-visual interactive soundscapes have been shown to play a key role in the understanding of public spaces [29]. The use of waterscapes, for example, can improve the comfort level of an environment [1]. Generative music tools have been shown to encourage community participation in cultural activities, providing real-time improvisation for users with a low-skill entry point [10]. The opportunity for generative audio feedback systems to facilitate even more meaningful interactions will therefore be furthered by a greater understanding of the human-machine interaction methods used to control such a system.

1.2 Project Aims

This report presents an experimental study that compared the user experience of interacting with a generative audio feedback system through use of hands compared to the full body, and also the experience with limited range of control versus continuous control of the output audio. The overall aim is to provide insight into the challenges of and opportunities for designing an interactive generative audio feedback system when using multiple gesture control methods. This can be broken down as follows:

1. **Movement-based generative audio feedback system:** To design a generative audio feedback system that uses body movement as control inputs.

2. **Compare control methods:** To compare the user enjoyment and intuitiveness of multiple control methods and their potential applications when used to control the audio feedback system.

3. **Real-time interaction:** To design a system that enables fast real-time interactions of 10 fps between input gestures and the audio feedback system.

4. **Gesture classification accuracy:** A highly accurate gesture-classification algorithm, in line with a 97% performance accuracy benchmark for single-person gesture recognition systems that have been used in interactive audio systems [27, 7].

5. **Modular System:** To design a system which allows for the addition of new sounds or instruments.

6. **Enjoyable sound generation:** Generate an audio feedback system that creates an enjoyable sound.

Chapter 2

Background Theory

The following section outlines technical approaches that have been used to track body movements, used in generative music systems, and machine learning algorithms that have been used as a tool for music interaction.

2.1 Human-Machine Interaction Technologies

2.1.1 Computer Vision

Computer vision has been used for pose estimation — a task that refers to inferring the spatial location of joints of the human body in 2D or 3D [35]. It can be categorised by two stages: detection and tracking. The detection stage involves the segmentation of the human from the background. Tracking involves the continual classification and tracking of key features of the human body, known as landmark tracking. Human pose estimation models typically use datasets that contain labelled instances of thousands of images of humans, such as COCO [30], MPII [2] and FLIC [40].

Recovering 3D single-pose estimation from 2D images using a single monocular RGB camera has been well documented [32, 56, 5] and the techniques used have been implemented in open-source models such as MediaPipe Pose [3]. Commercialised depth-sensing cameras such as Kinect [55] and Leap Motion [33] have also been used to great effect in research to fulfil 3D human gesture recognition. Although Leap Motion only has hand gesture recognition capabilities, as opposed to the full body.

Single and Multi-person Pose Estimation

Pose estimation can be classified as single or multi-person estimation. Single-person pose estimation (Figure 2.1) is generally more simple, and there are a multitude of pre-trained models that accurately estimate single-person poses in real-time from images that contain multiple people and can be embedded into user applications. Multi-person pose estimation can be approached using two approaches: top-down or bottom-up. Top-down multi-person pose estimation involves first detecting people in an image followed by applying single-person pose estimation to a cropped image of each detected person. Bottom-up detection first requires detecting all keypoints in an image and then grouping the keypoints into different people. Multi-person pose estimation is significantly limited by its computational complexity and hardware requirements. For accurate real-time response, successful multi-pose estimation algorithms are typically run on a GPU and employ between 4 to 5 RGB cameras (e.g., [13, 54].

Pre-trained Models

Popular pre-trained single-person pose estimation models include OpenPose [6], MediaPipe BlazePose [3], PoseNet [37], HRNet [44], AlphaPose [17] and Deep Pose [47]. MediaPipe, OpenPose and PoseNet have the advantage of an open API and suitability for CPU devices, giving users the flexibility to implement these architectures according to their specifications and on various hardware devices. Both OpenPose and MediaPipe offer real-time 3D single-person and 2D multi-person keypoint detection, however MediaPipe has greater real-time capability for CPU implementation [34] (10 fps for BlazePose Full and 31 fps for BlazePose Lite compared to 0.4 fps for OpenPose [3]). MediaPipe infers 33 3D keypoint landmarks compared to 17 for the OpenPose model. Additionally, MediaPipe offers a simple extension to MediaPipe Hollistic for simultaneous pose, hand and face landmark tracking.

2.1.2 Other Techniques

Data Glove

A data glove is a system that is frequently used for tracking and recognition tasks involving hand gestures. They are made up of an array of sensors and electronics for data acquisition, which record data related to the 3D hand motion of the user wearing the glove [11]. The sensors typically used in data gloves are inertial, magnetic and flex sensors. Data gloves that employ inertial sensors are widely used thanks to their low

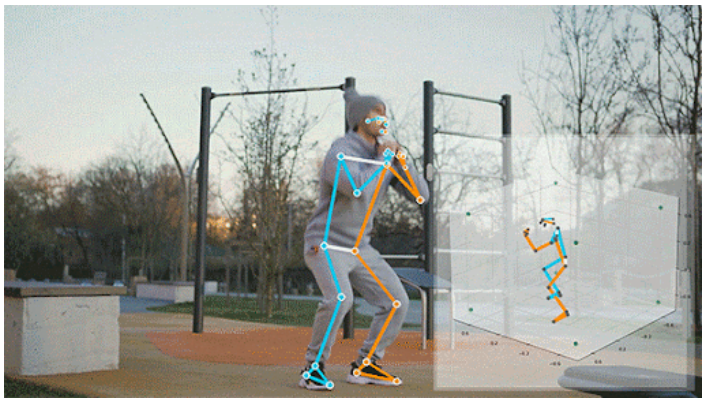


Figure 2.1: 3D pose estimation with MediaPipe BlazePose using 33 keypoints trained on COCO dataset [3].

cost and intuitive motion detection [16, 21]. The Mimu Glove is a commercial data glove that uses flex sensors to measure the bend of fingers [25] and is commonly used in composing and performing music using hand movement. Compared to inertial sensors, flex sensors are lighter and therefore have an improved user experience. Nonetheless, data gloves do not afford the same flexibility as contact-free computer vision techniques.

Electromyography

Electromyography is a measure of the electrical activity in response to muscle activity. Surface electromyography (sEMG) signals are captured by contact between electrodes and the skin. The EMG signal is then acquired by summing the action potentials discharged by muscle fibres in the operating region of the recording electrodes [26]. Myo is a commercially available gesture control armband that detects hand gestures using proprietary EMG sensors by measuring the stretching and contracting of forearm muscles responsible for hand movement. It has been widely adopted in both medical and creative applications [41, 46]. The cost of commercially available full body EMG technologies, however, greatly exceed the project budget.

2.2 Machine Learning as a Tool for Music Interaction

2.2.1 Generative Music Systems

Generative Models

Developments in generative adversarial networks (GANs) have enabled predictive models to produce music that is recognisable to listeners in both sound and feel to existing compositions. Music generation systems that utilise neural networks can be categorised into those that learn from raw audio to produce music that sounds like existing forms of music within set styles (e.g., WaveNet [36], WaveGAN [12] and GANSynth [14]), and symbolic models, such as MIDINet [53], that are trained on musical scores to produce music at note level. Symbolic models are widely regarded as inferior as they lack the ability to accurately represent the precise textural qualities (such as timbre, timing and volume) of a note played by a musician, and are often restricted to a particular instrument [9]. Generative music models that use generative adversarial networks are hugely limited by the training time and computational demands required. WaveGAN, for example, required a training time of 7.5 days when processed on a server with two NVIDIA Tesla V100 GPUs [52].

Granular Synthesis

Granular Synthesis [39] is a generative music technique that consists of segmenting sound samples into thousands of micro-acoustic events, known as grains, which are typically between 1 to 100 ms (Figure 2.2). Over time, the combination of thousands of grains can create animated and complex soundscapes. Unlike sample-based synthesis, which fails to abstract frequency-domain information, a grain is an appropriate depiction of musical sound as it contains information in both the time-domain (starting time, duration and envelope shape) and frequency-domain (waveform pitch and spectrum). The complexity of the sound produced depends on the changing combinations of many grains.

Granular synthesis has been combined with machine learning for interactive applications. A study intended to better understand human-computer interaction for digital musical instrument composition concluded that granular synthesis was a popular sound generation technique for its non-discrete exploratory features [19]. Another study integrated machine learning and granular synthesis, employing a classification algorithm

to affect the starting point of a granular synthesis module according to the image classification from a live video stream pointed to objects in a room such as postcards of different animals [31].

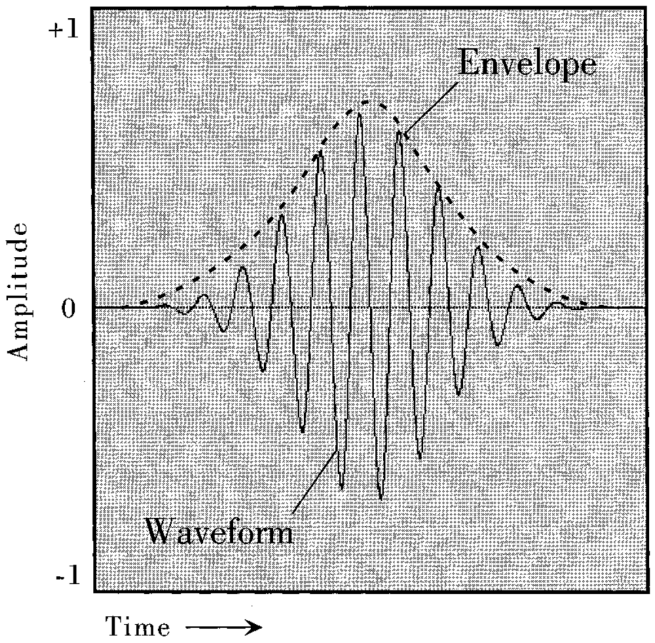


Figure 2.2: Grain waveform in time domain [39].

2.2.2 Training

A training dataset is generally representative data of the inputs that are to be seen by a machine learning model once implemented. Numerical features act as inputs to the system, and can be thought of as a way to describe the data. Algorithms generally perform best when exposed to large amounts of training data, however, obtaining large quantities of data for the novelty of the requirements, such as obtaining data for specific gestures can be difficult, where such knowledge is limited, unreliable or unavailable [18].

One method that is used to overcome this is transfer learning. Transfer learning allows the refinement of models trained on larger, more generalised datasets by substituting the final classification layer of a pre-trained model with a layer containing the desired classes for the application. This enables accurate models to be used without the need to collect an exhaustive training dataset. For example, a pre-trained model trained on hundreds of thousands of images of people for the purpose of human pose

estimation, could be repurposed to classify a tree pose for a yoga application, without the need to manually collect a large number of images of tree poses.

Interactive Machine Learning (IML) is a common method used in creative applications in which obtaining relevant training data is challenging. IML is the design of algorithms and intelligent user interface frameworks that enables machine learning through human interaction [15]. These systems enable flexibility in which the design specifications may be adaptive or exploratory, or where the required training data does not exist in pre-trained models.

2.2.3 Learning algorithms

Machine learning algorithms can be categorised by supervised, unsupervised and reinforcement learning, the use of which is dependent on their application.

Supervised Learning

Supervised learning algorithms are used to map one input domain to a different output domain. They require labelled input (or *training*) data and output (or *target*) data, known as *input-output pairs*. The algorithm learns from labelled training data to determine the likely output of new unseen input data. Supervised learning techniques include classification, regression and temporal modelling. Classification algorithms involve assigning a discrete class label to unseen inputs, for example, the ability to recognise certain physical gestures. Regression algorithms are used to produce continuous input-output mappings as a probability of the occurrence of the input-output pair. Regression models are therefore useful when implementing systems in which continuous outputs are desired. The Imitation Game [22] uses supervised learning by employing a feed-forward neural network trained to recognise various percussion instruments and playing techniques to control a robotic percussionist that either imitates what it has heard, initiates new sound material, or improvises based on an aspect of the musician's playing.

Temporal modelling techniques utilise a memory element to align incoming time-series data to template examples used to train the system. They are useful when the timescale of an input is imperative to the output of the system. For example, for a conductor, the time-taken for the same action to occur may dictate the tempo, or dynamics of a piece of music. One study uses an IMU and muscle sensing device for continuous gestural sound interaction based on Hidden Markov temporal modelling

learning methods, whilst another used Dynamic Time Warping for mapping conducting gestures tracked using a Kinect camera [43].

Unsupervised Learning

Unsupervised learning algorithms learn the internal structure of unlabelled input data. They can be used to identify clusters or discover patterns and structure within a training set, whose classification or type may be unknown to the designer. A common use of unsupervised learning methods is for raw audio or symbolic generative music models. Genetically Sonified Organisms [49] uses unsupervised machine learning methods to respond in a similar vocabulary to the natural environment and wildlife at the location where the physical agent is installed.

Reinforcement Learning

Reinforcement learning algorithms works by rewarding desired behaviours and/or associating penalties with undesired behaviours to achieve an optimal solution. Over many iterations, the agent learns to avoid generating negatively associated data. *Human-in-the-loop* music systems can employ reinforcement learning by continuously adapting the sound generation according to the specific interactions of the user. NOISA [45] is an intelligent human-in-the-loop system that estimates user engagement by monitoring their movements and facial expressions which act as control inputs to a generative music learning module.

Chapter 3

Methods

Given the project aims to explore a range of control methods using the full body, the use of multiple human-machine interaction technologies is both costly and time-inefficient. For this reason, computer-vision will be used for human-machine interfacing. Specifically, MediaPipe Pose will be used for human pose estimation to enable full body real-time tracking. The specifications for systems that use machine learning as a creative tool are uncertain during the early stages of the design process, making prototyping and experimentation imperative for designers [18]. The flexibility afforded by interactive supervised machine learning algorithms enables working prototypes to be implemented in a short period of time which can be iteratively refined by fine-tuning model parameters and adjusting training data. For this reason, supervised learning will be used. The computational demands of generating music from both raw audio and symbolic models make them inappropriate given the project time scale. Moreover, as they essentially act as "black boxes" and use imitation as an optimisation objective, their applicability for human-computer interaction tasks, in which a key objective is to challenge one's familiar creative habits through exploration, is limited [23]. Granular synthesis will therefore be used, as it facilitates both real-time human-computer interaction and exploratory music generation. Figure 3.1 illustrates the design process, which has been broken down into three distinct subsystems described below.

1. **Pose Estimation:** Real-time inference of 3D joint coordinates of the identified person.

2. **Gesture Recognition:** Real-time classification and regression of incoming gestures.

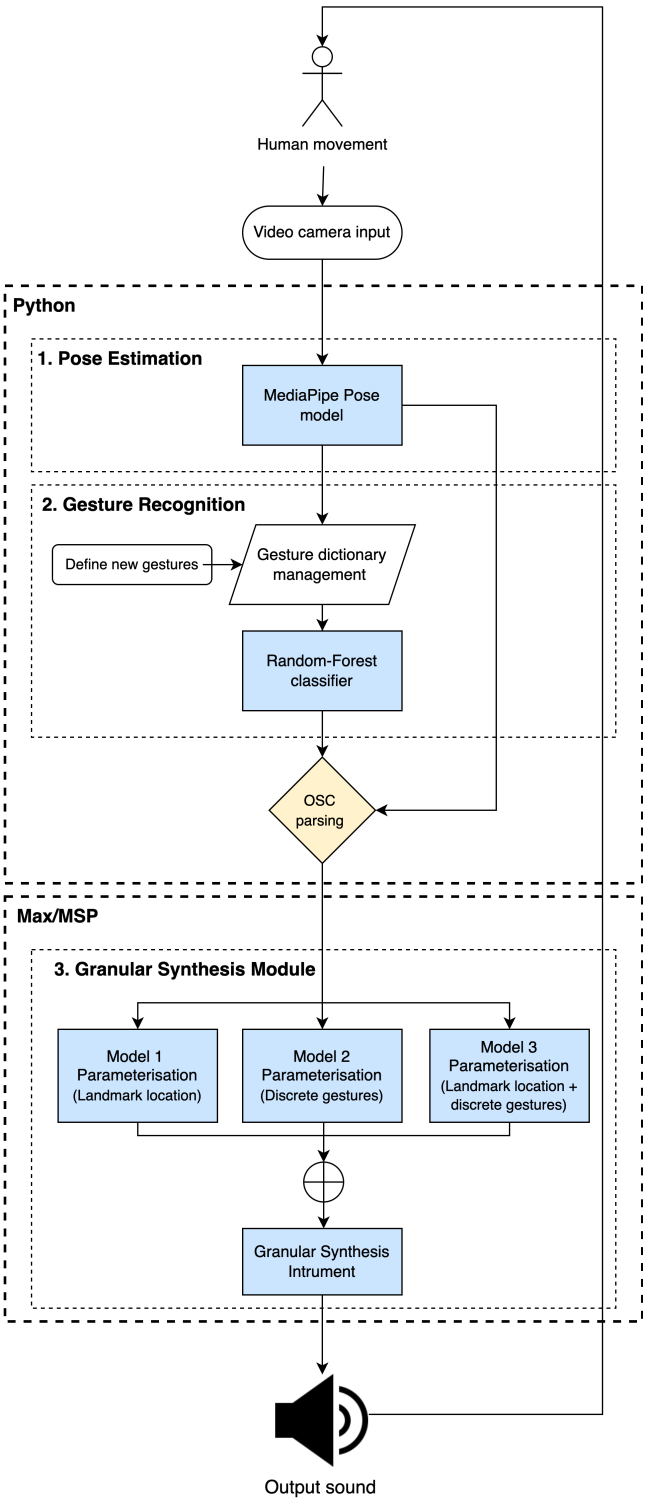


Figure 3.1: Schematic of interactions between the subsystems used in the generative music feedback system that uses human movement as control inputs.

3. **Granular Synthesis Module:** A generative sound module designed to map discrete gestures and pose landmark locations to simple musical control parameters.

3.1 Pose Estimation

The pose estimation model was implemented on a Macbook 2.4 GHz Quad-Core Intel i5 and used the MediaPipe Pose model, built upon *BlazePose* [3], to track 33 landmarks in 3D, using a single webcam. Implementing a pre-trained model on a laptop afforded flexibility in determining key hardware requirements prior to hardware installation.

3.1.1 Parameters

The model detects the person most prominent in the first images and subsequently tracks those landmarks without invoking another detection, thus reducing computational load. The algorithm is then set to filter pose landmarks over multiple images to reduce jitter.

- **Minimum Detection Confidence:** The minimum confidence value for the detection to be successful, ranging between 0-1. Set to 0.5.
- **Minimum Tracking Confidence:** The minimum tracking confidence value for the landmarks to be successfully tracked, ranging between 0-1. Set to 0.5.
- **Model Complexity:** The complexity of the model: 0, 1, or 2. Set to 2.

3.1.2 Output

The pose estimation model outputs three normalised landmark coordinates, x,y, and z and a visibility value which indicates the probability of the landmark being present and not occluded.

Real-world Landmarks

A list of x, y and z landmark coordinates are output as real-world coordinates with respect to the centre of the hips. The use of real-world coordinates rather than image coordinates enabled simple calibration to user height, irrespective of their location in the camera's field of view. Additionally, when training the machine learning model (as in section 3.2), real-world coordinates enabled greater flexibility when collecting training data. This was because subjects were free to move so could practice the

gesture naturally, without the model learning their location with respect to the image as a key feature of the gesture class.

3.1.3 Hardware Requirements

To inform the camera selection, the effect of image resolution and model complexity on frame rate and landmark visibility was tested on a video of a dancer which was down-sampled from 1080p to 280p. The video resolution impacted the output frame rate, which varied greatly depending on the model complexity, as seen by 3.2. A frame rate of 20.1 fps was achieved for a high video resolution of 720p using a model complexity of 1. For video resolutions greater than 480p, the frame rate was approximately half when using a model complexity of 2 compared to a model complexity of 1. A video resolution of 1080p yielded a frame rate of 8.3 fps which was insufficient to satisfy the project aim frame rate of 10 fps. Indeed, the maximum frame rate was only 10.9 fps for a model complexity of 2. Given the computational demands of the gesture recognition and sound modules that follow, the average frame rate would be at risk of falling below 10 fps. A model complexity of 1 was therefore used to fulfil the pose estimation task. Figure 3.3 shows that video resolutions of 640p and greater had minimal effect on landmark visibility. For this reason a 720p camera was sufficient to fulfill the pose estimation task, achieving a frame rate of 20.1 fps. An ASUS C3 720p web-camera was used on this basis.

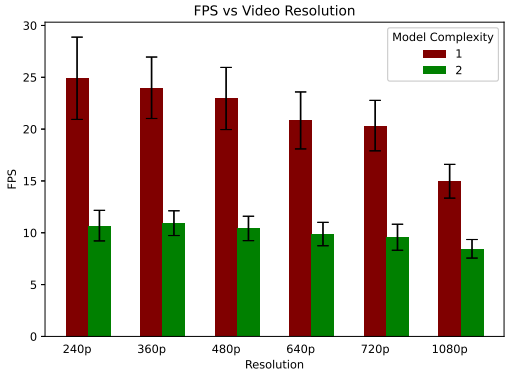


Figure 3.2: Graph showing the effect of input video resolution and model complexity on the average camera frame rate using the MediaPipe Pose over 100 frames.

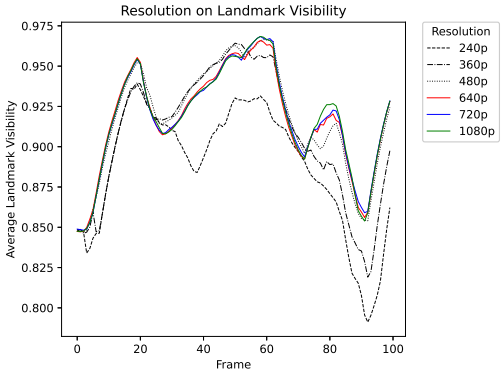


Figure 3.3: Graph showing the effect of video resolution on the landmark visibility of a person inferred from the MediaPipe Pose Estimation model over 100 frames.

3.2 Gesture Recognition

3.2.1 Gesture Selection

Laban Movement Analysis [28] served as the foundation for the classification and feature analysis. The movement types can be characterised by the Effort notation, which consist of four components, with each component composed of two underlying principles that describe its expression: *direction* (direct or indirect), *weight* (heavy or light), *speed* (sudden or sustained), and *flow* (bound or free). The combination of the four components and their elements constitute the Eight Efforts (Figure 3.4): *wring*, *press*, *flick*, *dab*, *glide*, *float*, *punch*, and *slash*. Given that the machine learning model does not account for feature extraction in the time domain, the selection of gestures that were contrasting in landmark positions and also represented contrasting Laban movements was key to enabling an intuitive gesture-sound mapping when interfacing with the sound generation module. Four gestures: **flick**, **float**, **kick** and **punch**, were selected to represent the Laban efforts and train the learning module.

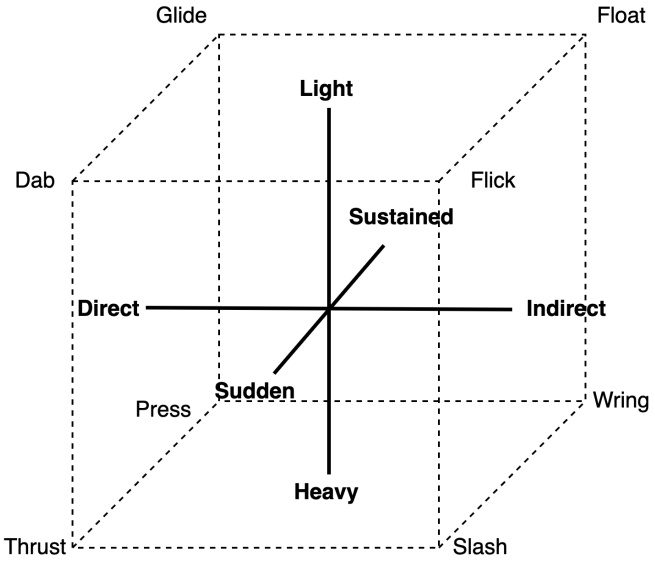
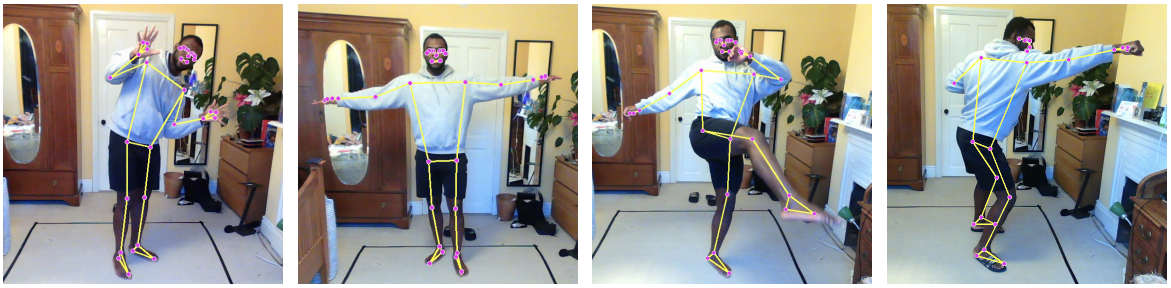


Figure 3.4: Laban Movement Analysis axes. The components of the Effort notation are in bold.

3.2.2 Generating Training Data

The pose estimation model was used to infer 33 landmarks for five users to collect training data for each gesture class (flick, float, kick and punch (Figure 3.6) and an

additional neutral pose. For each gesture class, the user was free to move around a 2 x 2 m performance area for one minute. A five second timer signalled to users when to be in the relevant gesture position. At the end of the timer a stream of 10 images were captured and the corresponding landmark data was added to a training script. This gave a total of 3000 training images, and a total of 396,000 training datapoints (3000 x 33 landmarks x 4 coordinates (x,y,z,visibility)). Following initial data collection, it was apparent that too few samples in crouching positions had been obtained. The iterative nature of collecting training data through user input-tracking enabled greater flexibility and the ability to amend training data. A further 800 training images were captured for punch and flick gestures in crouching positions.



(a) Flick (b) Float (c) Kick (d) Punch)

Figure 3.6: Example samples of a user performing each gesture class during the training phase for which the landmark coordinates inferred from pose estimation are used as training data for the gesture recognition algorithm.

3.2.3 Random Forest Regression

Random forest is a supervised learning technique that utilises ensemble methods to obtain better predictive performance than those performed solely using decision trees. They combine two processes, *bootstrapping* and *aggregation* into a process known as *bagging*. Bootstrapping ensures that the same data is not being used for every decision tree by creating multiple new random datasets from the original training dataset. Each new training dataset contains the same number of rows as the original. A decision tree is then independently trained on each new dataset, with a random subset of features of each dataset used to train the corresponding decision tree. The use of random feature selection reduces the correlation between the trees, since using the same features for each tree would result in the same decision nodes across multiple trees, thus increasing the variance. Whilst the predictions of a single tree may be highly sensitive to noise in its training set, as some trees are trained on less significant features, the average across

multiple trees is less affected by noise as the trees are not strongly correlated. The overall variance of the model is therefore reduced through bootstrapping. The number of random features selected was set to 11, i.e., the square root of the total number of features, 132. A prediction is made by passing the features through each decision tree, with each classifier contributing a single vote and the outcome a combination of those votes (known as *aggregation*). By taking the average across the decision trees at the aggregation stage, a regression for each gesture class was obtained.

3.2.4 Gesture Recognition Performance

The random forest classifier was trained on 70% of the total training data, with 30% of the data used for validation. The data was stratified to preserve the class ratios in both the training and validation dataset. Using simple cross-validation and stratification, the random forest classifier achieved a total performance accuracy of 98.6% , with 969 correctly classified as true positive (Figure 3.7). The induced error was split equally (16 false positives and 16 false negatives), with punch and float classes each constituting 43.75% of false positives and false negatives, respectively.

The performance of the random forest classifier was significantly better than linear regression models such as logistic regression and ridge classification, which yielded respective performance accuracies of 89.3% and 86.7%. Random forest classifiers are built using decision trees (as described in section 3.2.3), which split the training dataset into several subsets prior to making a prediction. By contrast, linear models, which assume a linear relationship between input and output data, tend to under-fit the model, as a straight line fails to adequately fit complex datasets.

3.3 Granular Synthesis Module

The granular synthesis module was implemented in MAX/MSP — a programming environment used for real-time audio and visual synthesis processing. To design a flexible and highly accurate granular synthesiser, with high polyphonic routing capability (i.e., the ability to route multiple instruments and voices simultaneously), the first step was to implement a grain generator (see Appendix A for MAX/MSP implementation patches).

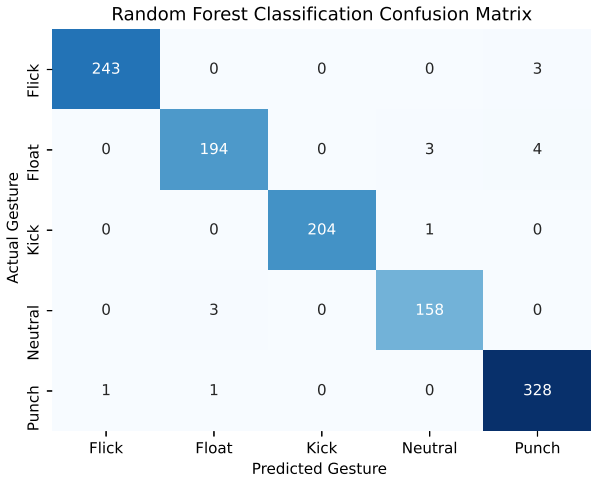


Figure 3.7: Confusion matrix showing classification of gesture recognition algorithm against labelled gesture data. True positives are the matrix diagonals, false positives are the columns (excluding matrix diagonal) and false negatives are the rows (excluding matrix diagonals)

3.3.1 Grain Generator

The grain generator is a digital synthesis instrument that serves as the centrepiece for granular synthesis. An envelope sets the amplitude of a wavetable oscillator, whose output is sent to a spatial panner of N channels, as shown by Figure 3.8. The specific properties of the micro time scale mean adjusting the grain envelope and grain duration significantly contributes to the spectrum of the resultant signal. The grain generator parameters *start*, *duration*, *pitch*, *density*, *spatial location*, *envelope* and *buffer*, are described below:

- **Sound Buffer:** The audio sample used from which grain waveforms are extracted
- **Start, x_0:** The time position in the buffer which serves as the start of the grain waveform in milliseconds
- **Duration, x_d:** The length in milliseconds of the grain waveform. A negative duration results in a reverse reading of the grain waveform.
- **Pitch, k_p:** The transposition factor of the grain waveform. This effectively adjusts the speed at which the grain waveform is read by multiplying the grain duration. For example, a factor of 1 corresponds to the original speed and pitch, 0.5 to an octave below, and 2 to an octave above the original speed.

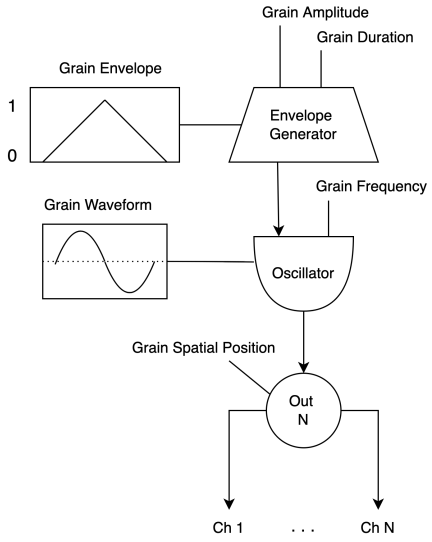


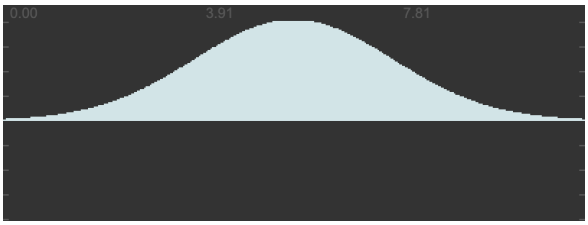
Figure 3.8: A simple grain generator used in granular synthesis.

- **Grain Density, ρ:** The number of grains omitted per second
- **Spatial Location:** The channel output distribution of each grain or cloud
- **Amplitude Envelope:** The resultant amplitude of the grain waveform when convolved with an envelope

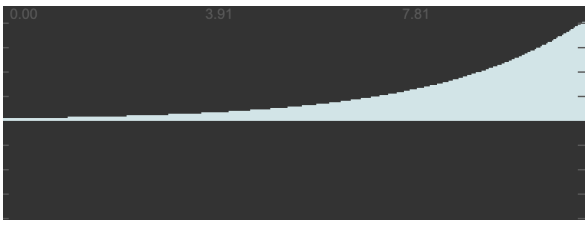
Sound buffers enable granulation of the most varied sound materials. They can be synthetic, such as using sinusoidal signals, or sample-based, which may be extracts from environmental soundscapes, or music compositions. Prior to instrument selection, an arbitrary sample was used as the buffer to facilitate the implementation of the grain generator. The *line~* object was used to scan through an audio buffer (set in *play~*) from x_0 to $x_0 + x_d$ in kx_d milliseconds. The output signal was then multiplied by an envelope of duration x_d and sent to a spatial panner which randomly dispersed each grain to the left or right output (N=*2*) of 2-channel stereo speakers. The effect of spatial panning is one of creating depth. If each grain had the same spatial position, i.e., a monaural cloud, it would be spatially flat. By scattering each of the grains to a unique location, the sound simulated a three-dimensional spatial morphology.

The amplitude envelopes assigned to each grain were assigned to buffers. The advantage of reading envelopes in buffers was one of increasing computational efficiency. The mathematical operations that define common envelope shapes such as the Gaussian are computationally expensive, and when triggered at high densities, may compromise the

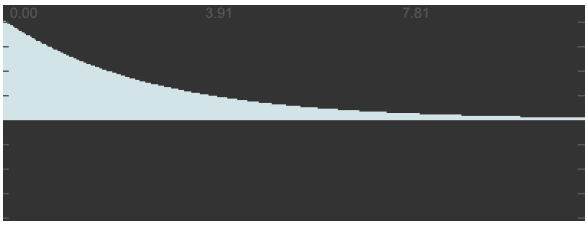
real-time capability of the grain generator. By pre-defining each envelope in buffers, computational efficiency was improved when reading in real-time. Five envelope buffers were implemented and their respective waveforms are shown in Figure 3.9. Each envelope buffer was quantised to 512 samples.



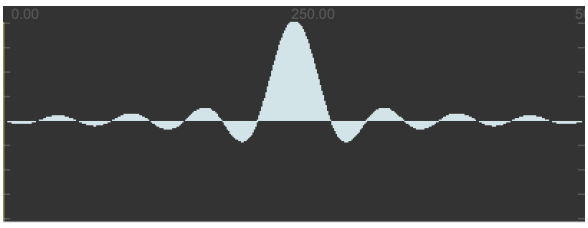
(a) Gaussian



(b) Expodec



(c) Rexpodec



(d) Sinc

Figure 3.9: Envelope buffers used to shape the grain waveform generated in the granular synthesis module.

3.3.2 Grain Triggering

The grains must be triggered upon implementing the grain generator. The frequency of triggering determines the grain density. To enable multiple instances of the grain

generator, therefore allowing simultaneous triggering of grains to create textured clouds, the grain generator was instantiated using the *poly~* object. Each instance of the grain generator, known as a *voice*, was sent its respective parameter values via a packed list. Each instrument consisted of 8 voices.

The internal message response of Max/MSP is not well adapted to high-frequency messages greater than 1000 Hz, meaning at high grain densities there were timing irregularities. To bypass this, the grains were triggered by producing an impulse at the zero crossing of a sine wave, enabling high density grain generation. The output of the zero-crossing impulse produced a note on-off pair which triggered the grain generator. The note-on turned the DSP on whilst the note-off turned the DSP off by muting the *poly~* instance once the grain had been generated, thus improving the computational efficiency of the granular synthesis module.

Synchronous and quasi-synchronous grain triggering methods were implemented. Synchronous granular synthesis involves triggering grains at regular intervals. In quasi-synchronous granular synthesis, grains follow one another at irregular intervals according to a normally distributed random variable with boundaries set by the user.

Quasi-Synchronous Triggering Using Stochastic Algorithm

A stochastic algorithm inspired by Brownian motion [48], which describes the motion of particles in fluids following inter-particle collisions, was implemented to give the grains a perceived flow over time. The algorithm takes two parameters which represent the lower and upper limits of the output grain density, and outputs a random number between the two limits. This number is at a distance from the previously output grain density determined by a variance factor, k_d. When $k_d = 0$, the resultant grain density is the same as the previous grain density. When $k_d = 1$ the algorithm generates a random number between the lower and upper bound that has no relation to the previous grain density.

3.3.3 Instrument Design

Three instruments were developed using the granular synthesis module. Each instrument was designed to be contrasting to the others in sound, and represent at least one of the Laban movements described in Figure 3.4. This was achieved by designing instruments which each characterised different musical elements. The first instrument, a bubble

generator, represented rhythm. The second, a harp, represented harmony. The third, ocean waves, represented dynamics and articulation. Pitch, tempo and gain were chosen as the instrument elements that could be varied by the user due to their common familiarity compared to other musical techniques that may require a greater level of prior knowledge. To maintain a sense of unpredictability and continuous generation, the exact parameter value was randomly selected within upper and lower limits determined by the user.

Bubble Generator

A bubble generator instrument used a 100 ms synthetic signal buffer convolved with a variable sinc envelope (as in Figure 3.9d). Its pitch and tempo were determined by the grain duration, the envelope buffer and the quasi-synchronous grain density granular synthesis parameters (Table 3.1). The constant granular synthesis parameters were the grain start ($x_0 = 0$) and the pitch ratio ($k_p = 1$).

Table 3.1: Variable granular synthesis parameters for bubble instrument.

Musical Element	Granular Synthesis Parameter	Effect	Parameter Value
Pitch	Grain Duration	Low	3 ms
	Grain Envelope		Sinc(16)
		High	38 ms
			Sinc(8)
Tempo	Grain Density	Slow	0.5 g/s
		Fast	10 g/s

Harp

Harmony can be a difficult task to fulfil in granular synthesis, as is generally the case with generative music. It often produces discordant clashing sounds, and when using longer samples, it can be hard to distinguish any form of harmony whatsoever as the grain selection space within the sample is large. The harp was chosen as an instrument that is harmonically pleasant and one that enabled flexibility and fluidity in its sound and feel. It was a sample-based granular synthesis instrument, using a 11.71 second harp sequence uploaded to its buffer. The grain start and grain duration were the variable synthesis parameters used to adjust the pitch and tempo, respectively (Table 3.2). The grain pitch ratio of each voice was set randomly between three octaves,

creating layered and highly textured outputs. Due to the length and complexity of the harp sample (almost 120,000 times larger than the bubble buffer), controlling the tempo and pitch of the instrument was not a simple process and produced additional sound effects. To achieve a low pitch, the grains were chosen closer to the end of the buffer, at around 7500 ms, where a lower register existed in the sample. To achieve a higher pitch, grains were chosen from the first few hundred milliseconds of the buffer. The texture of the harp sequence adjusted with the grain duration. At low grain duration ($100 < x_d \leq 300$), the perceived sound was one of pulsating, with fast and light harmonic sequences contained within each pulse. At higher grain duration, the perceived sound was light in texture, but longer and more akin to floating or gliding movements.

Table 3.2: Variable granular synthesis parameters for harp instrument

Musical Element	Granular Synthesis Parameter	Effect	Parameter Value
Pitch	Grain Start	Low	$7100 < x_0 \leq 8500$
		High	$500 < x_0 \leq 1500$
Tempo	Grain Duration	Low	$4000 < x_d \leq 8500$
		High	$100 < x_d \leq 300$

Waves

The sound of ocean waves represented dynamics and articulation, which when mapped to the Laban gestures, would be activated by punching motion. To achieve this, the pitch scale factor and the grain density parameters were varied of an ocean wave sound buffer. At higher grain densities (between 1.5 and 2 grains per second) and lower pitch ratios (0.5), the perceived sound was one of violent waves crashing. The grains at these thresholds were filtered using a rexpodec filter (Figure 3.9c). The sound of waves lightly breaking was achieved by increasing the pitch transposition factor to between 2 and 3 and reducing the grain density to between 0.2 and 0.5 grains per second, with each grain Gaussian filtered.

3.4 Interfacing Pose Model and Granular Synthesis Module

To understand the preferred methods for controlling the instruments, three scenarios were implemented. The first scenario looked at the continuous control of the instrument

parameters (pitch and tempo) by using raw landmark wrist location. The second scenario looked at constrained control of the same instrument parameters by using discrete gestures to control the instrument parameters. The third scenario involved continuous control using a combination of landmark data and gestures to control the instruments. The landmark and gesture recognition data had to be parsed to the sound module and pre-processed before each scenario was carried out.

3.4.1 Parsing OSC Data and Pre-Processing

Parsing OSC Data

A networking module was developed in Python to communicate real-time OSC landmark and gesture recognition data to a server patch in Max/MSP, and simultaneously receive data from Max to facilitate the graphical analysis of particular Max objects when selecting filters, and to record data retrieved from the user study.

Landmark Filtering

The slide object in Max/MSP was used to filter the incoming landmark data. It functioned as a low-pass filter by smoothing continuous values logarithmically between two parameter values (*slide-up* and *slide-down*), thus removing any sudden fluctuations and jitters incurred from the pose detection model. Figure 3.10 shows the effect of three slide filters as a user moved their arm in a radial motion from straight down by their side to the side (x-y plane) and in front of them (y-z plane). The response time decreases significantly for the slide 10 filter, whereas for a slide 2 filter it closely approximates the original raw data. For sudden movements such as punch or kick, a slide 2 filter was used to satisfy the desired fast response. For slower sustained movements, such as floating and gliding, a slower response yielding a smoother transition between sounds was obtained using a slide 5 or 10 filter which was particularly effective for fading out the wave instrument. For the gesture classification, a slide-up parameter of 2 was used for the punch and kick classes, slide-up 5 was used for the flicking class and slide-up 10 was used for the gliding class. Fading-out was achieved by increasing the slide-down parameter to 10.

Calibration

To enable each user to ascertain the same parameter boundaries when using the instruments, the algorithm was calibrated to the user's height by averaging the distance

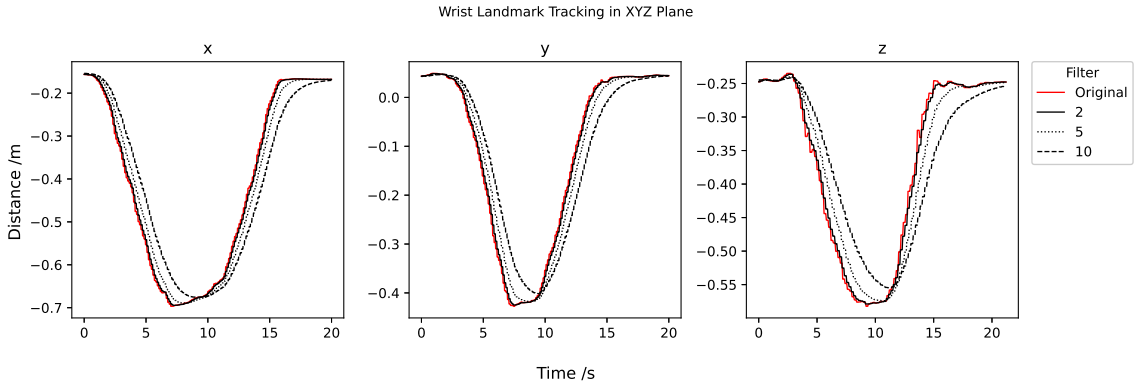


Figure 3.10: Effect of Max/MSP slide filtering on tracking wrist landmark when moving arms by side outwards (x,y) and in front (z).

between the eye and heel landmarks over the first 50 samples of them standing in the designated performance area.

3.4.2 Scenario 1: Instrument Selection using Body Location and Instrument Parameter Control Using Landmark Data

The first scenario used wrist landmark data to control each instrument. The performance area was divided into two sections: bubbles and harps. An instrument was activated by measuring the average x location of the user in the performance area.

Tempo

The tempo of the instrument was mapped to the calibrated horizontal distance between the left and right wrists, Δw_x (Figure 3.11a).

$$tempo = f(\Delta w_x)$$

For the bubble generator, this distance controlled the upper and lower limits of the grain density parameter. The upper limit, $\rho_{2,bubble}$, was set by the distance between the wrists scaled between 0.5 and 10 grains per second. The lower limit, $\rho_{1,bubble}$, was a function of the upper limit and the unscaled distance between each wrist,

$$\rho_{1,bubble} = \rho_{2,bubble} \cdot e^{-\Delta w_x}$$

(a) Tempo (b) Pitch (c) Gain

Figure 3.12: Scenario 1 parameter control reference photos for controlling tempo, pitch, gain of generative sound module using wrist location.

For the harp sound, this distance controlled the grain duration limits. The lower limit of the grain duration, $x_{d,1,harp}$, was obtained by scaling the wrist distance between 180 and 6500 and the upper grain duration limit, $x_{2,d,harp}$ was a function of the lower limit

$$x_{d,2,harp} = x_{d,1,harp} \cdot e^{-\Delta w_x/2} + x_{d,1,harp}$$

Pitch

The instrument pitch was mapped to the height of the left wrist, $w_{l,y}$, such that raising the left hand position corresponded to a higher pitch (Figure 3.11b).

For the bubble generator, the pitch was determined by the grain duration and the sinc envelope shape. The left wrist location was scaled between 16 and 5 as the x-value for the sinc envelope, and scaled between 3 and 18 to form the lower boundary for the grain duration. A constant ($k = 23$) was added to form the upper boundary of the grain duration,

$$x_{d,2,bubble} = x_{d,1,bubble} + 23$$

To control the pitch of the harp, the left wrist location was mapped to the lower

boundary of the start location by scaling between 7100 and 500 and the upper start boundary was offset by 1000 ms from the lower boundary,

$$x_{0,2,harp} = x_{0,1,harp} + 1000$$

Gain

The gain for each instrument was controlled by the y-coordinate of the right wrist (Figure 3.11c), where a higher right hand corresponded to a higher instrument volume. By taking the y-coordinate as the linear amplitude, the gain, A, was obtained as a function of the linear amplitude, A_l, and its decibel conversion A_{db}, such that

$$A = A_{db} \cdot e^{-2 \cdot A_l}$$

3.4.3 Scenario 2: Instrument Selection using Body Location and Instrument Parameter Control using Discrete Gestures

(a) Pitch (low) (b) Pitch (high) (c) Tempo (low) (d) Pitch (high)

Figure 3.13: Scenario 2 parameter control reference photos for controlling tempo and pitch using discrete gestures: (a) neutral gesture is low pitch, (b) float gesture is high pitch, (c) two hands in front is low tempo and (d) one hand in front is high tempo.

Using the gesture recognition algorithm described in Section 3.2, four discrete gestures were mapped to the pitch and tempo of each instrument. The purpose of scenario 2 was to compare the user experience when restricting the parameter control space. Rather than being able to continuously adjust the pitch level or tempo between two limits, the user could control only one parameter at any given time, and this was limited to either *high* or *low*. The probability of the detected gesture corresponded to the magnitude of each instrument parameter (for example, a higher probability of the floating class

resulted in a higher pitch). The gestures were scaled to the same parameter thresholds as those described in scenario 1, and the bubble and harp instruments were again activated by the average landmark x-coordinates. Given the restricted control space, the user was no longer able to control the gain of the instrument. The gain amplification of each instrument, A, was a function of the probability of the detected class, P_{Max},

$$A = A_{db} \cdot e^{-4 \cdot P_{Max}}$$

Pitch

Pitch was controlled by the neutral and float classification gestures. A neutral position (Figure 3.13a), corresponded to a low pitch, whilst a floating position (Figure 3.13b) corresponded to a high pitch.

Tempo

Tempo was controlled by the push and punch classification gestures. A push gesture, i.e., two arms in front (Figure 3.13c), corresponded to a low tempo, whilst a punch gesture (Figure 3.13d) corresponded to a high tempo.

3.4.4 Scenario 3: Instrument Selection using Gestures and Instrument Parameter Control using Landmarks

(a) Bubbles = Flick (b) Harp = Float (c) Wave = Kick/Punch

Figure 3.14: Scenario 3 gestures used to activate associated instrument and the parameter control within each gesture. The green axis is tempo, the red axis is pitch. (a) the bubble generator is activated by flick gesture and hand position determines the pitch and tempo; (b) the float gesture activates harp and tempo is determined by the rotation of the arms; (c) waves are activated by kick and punch gestures.

Unlike in scenario 1 and 2, in which only one instrument could be activated at once, in scenario 3, all three instruments were simultaneously active, with the loudness of one instrument over the others dictated by the recognition of the gesture associated with that instrument. In doing so, multiple instruments were able to exist simultaneously creating a rich soundscape consisting of all the instruments. For the remainder of this section, the term "activate" will refer to amplifying the gain of one instrument compared to the other instruments.

The gain of each instrument , A_I, was determined by the probability of its associated gesture, $P(G)$. The harp instrument was activated by the floating gesture. The bubble instrument was activated by the flicking gesture and the introduction of the ocean wave instrument was activated by punching and kicking movements. If the associated gesture was the most prominent gesture (i.e., highest probability compared to all the other gesture classes ($P(G) = P_{Max}$), the instrument gain was amplified by a factor of 2. If a gesture not associated with the instrument was most prominent, the gain of the instrument was amplified by a factor of 0.6.

$$A_I = \begin{cases} 2 \cdot A_G & P(G) = P_{Max} \\ 0.6 \cdot A_G & otherwise \end{cases}$$

Each instrument had a neutral state, such that when the instrument was not in its active state, its control parameters (tempo and pitch) varied between a mid-range.

Bubbles: Flicking

The bubble instrument was activated by the flick gesture (Figure 3.14a). When a flick was detected, the average landmark location of the wrists was mapped to a two-dimensional tempo-pitch axis. The horizontal axis was the tempo, with the bubbles generating at a faster rate as the average wrist location moved from left to right. The vertical axis corresponded to the pitch so the bubble pitch would increase as the wrists moved from a low to a high position.

Harp: Floating

The harp was activated by the float gesture. The angle of arm rotation then determined the texture of the harp (Figure 3.14b). The angular rotation was extracted by measuring the angle between the shoulder and wrist landmarks and mapping it to the grain duration of the harp. As the angular rotation increased from 0° to ±90° the grain

duration decreased from a long grain duration, corresponding to a calm and drawn out harp sequence, to a short grain duration resulting in a faster, pulsating sequence of sounds. The pitch of the harp (determined by the grain start) was controlled by the average x-location of the user.

Wave: Kicking/Punching

The sound of waves were activated by the punch and kick gestures (Figure 3.14c). There was limited user control of the output sound for the wave instrument. The texture of the waves was also determined by the probability of the gesture classification. This probability affected the grain density and pitch transposition ratio granular synthesis parameters of the wave instrument, with a higher probability corresponding to a louder, more violent sounding wave. The granular synthesis parameters are as those described in Section 3.3.3.

Chapter 4

User Study

This section presents a user study designed to assess three main aspects of the scenarios described in Section 3.4: the controllability when using discrete gestures compared to continuous landmark tracking to control each instrument, the enjoyment of each scenario, and the use of generative music as a tool for public interaction. 18 participants with a range of ages, physical and mental abilities, and experience in sound took part in the study. Following a short demonstration of how to use the instruments in each scenario, the participant had a maximum of ten minutes to interact with the instruments in that scenario [1].

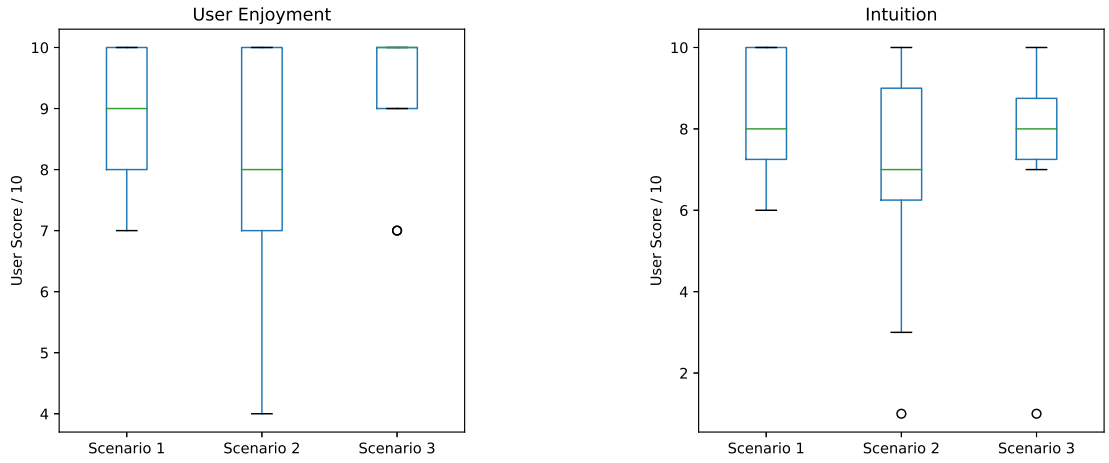
4.1 User Experience

4.1.1 Enjoyment and Intuition

Enjoyment

The participants were asked to rank the enjoyment of each scenario out of 10 (Figure 4.1a). All three scenarios were enjoyed, however scenario 3 ranked highest, with a median score of 10 and only two participants scoring it below 9. Scenario 2 ranked second, with a median score of 9 and a low score of 7. Scenario 2 was the least enjoyed, with a median score of 8 and a low score of 4, and only one person ranking it as the most enjoyable scenario. One reason for the discrepancy between the enjoyment of scenario 2 and scenarios 1 and 3 was the reduced parameter control. In scenario 2, only one instrument parameter could be controlled at a time, and the control within the chosen parameter was discrete, rather than continuous. This is reflected in Figure 4.2,

[1] Scenario 1 user study example video



(a) (b)

Figure 4.1: Graphical results from user study showing (a) enjoyment and (b) intuition, of the three control methods described in Section 3.4. For both user enjoyment and intuition, scenario 2 which uses discrete gestures as instrument control parameters scores lower than continuous parameter control (Scenario 1 and 3). The user enjoyment for activating instruments with gestures (Scenario 3) was the most enjoyable but less intuitive than using hand location (Scenario 1).

which shows a far greater number of participants with the opinion that the number of instrument control parameters was too low (7/18). Comparatively, 17 participants in scenario 1, and 15 in scenario 3, thought there was the right amount of control. A common reason given for the enjoyment of scenario 3 was the ability to use multiple instruments simultaneously. One user commented that the ability to play multiple instruments simultaneously "felt more like making music versus playing an instrument", whilst another said they "liked the way the sounds overlapped." Further exploration of these results is discussed in Section 5.

Intuition

Although scenario 3 was the most enjoyed experience, this did not translate to feeling that it was the most intuitive, as scenario 1 was deemed the most intuitive (Figure 4.1b). Both scenario 1 and 3 had a median score of 8, while scenario 2 had a median of 7 and a far greater distribution of results, with a low score of 3. A user who suffered from dyspraxia commented that scenario 2 "was really confusing" and it was "hard to grasp the gestures." The distinction between the intuition of scenario 1 and 3 was small, as some participants found using simple hand movements to control the

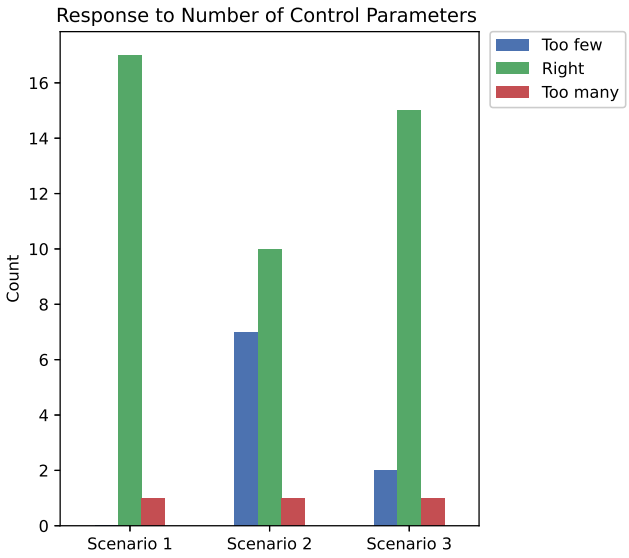


Figure 4.2: User study response to the number of instrument control parameters for the different control methods. 7 participants thought the number of control parameters using discrete gestures as a control mechanism (Scenario 2) was too few.

same parameters for each instrument (Scenario 1) more intuitive, whilst others felt the relation of discrete gestures to instruments was more intuitive (scenario 3).

4.1.2 Response to Sound Generation

Results showed a split in participant's favourite instruments between the harp and bubble generator. This split was consistent with the time spent on each instrument (Figure 4.3), which also shows a greater amount of time spent on scenario 1 (7 minutes 36 seconds) than scenario 2 (6 minutes 50 seconds) per user. Generally, participants enjoyed the harp sound for its depth and texture and "warmth of sound." By comparison, those who chose the bubbles gave their greater control over its parameters as the reason for this preference, with it being described as "the most directly interactive". The addition of the wave instrument for scenario 3 was positively received, with three users rating it their favourite instrument. One user said they "liked its immediate reactivity." The response to the unpredictability of the generated sound was also positive (Figure 4.4), with 14 participants thoroughly enjoying the unpredictability (score of 5), and only 2 participants scoring below 4.

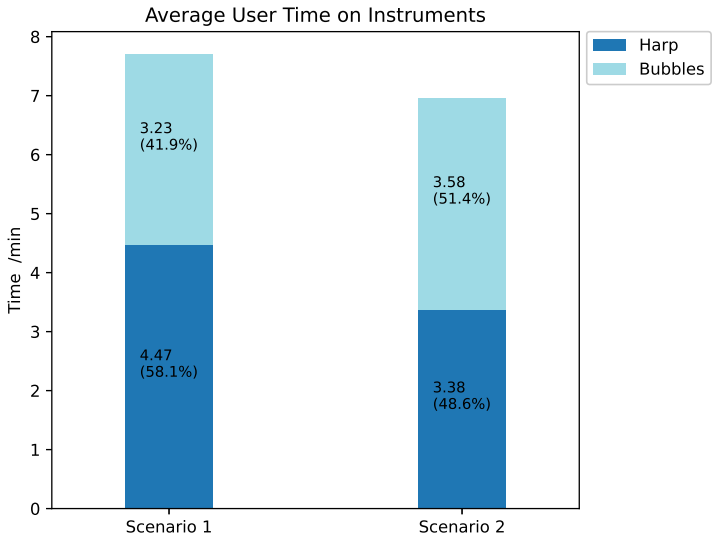


Figure 4.3: Average user time spent interacting with generative audio feedback system when using continuous control by wrist-location (Scenario 1) or discrete control using gestures (Scenario 2).

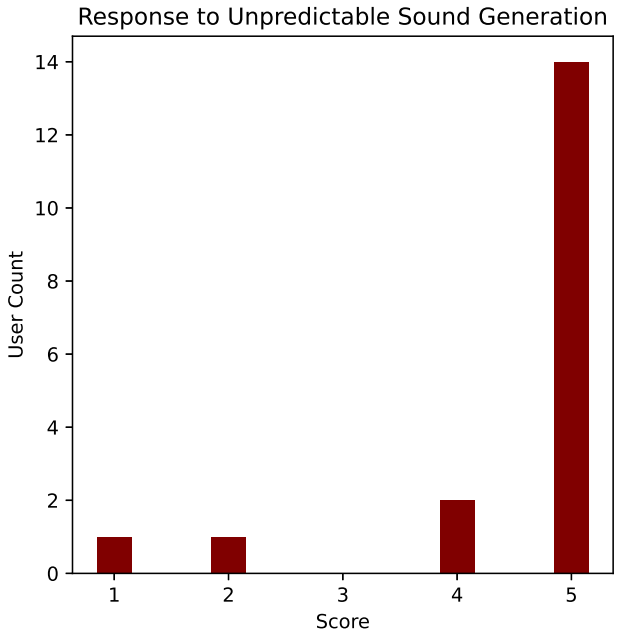


Figure 4.4: User response to enjoyment of sound generation. 88% of users scored 4 or above showing success of generative sound module.

Chapter 5

Discussion and Conclusion

This report has described the design and implementation of a digital instrument whose parameters are determined by human gestures and location. The system design consists of the following three subsystems. A pose estimation subsystem extracted skeletal landmark coordinates from an individual. A gesture recognition algorithm was used to recognise five gestures derived from the Laban Movements analysis. A granular synthesis sound module then mapped incoming pose and gesture data to control three digital instruments that represented contrasting musical elements in rhythm, harmony and texture. A user study was carried out to identify the user experience when using three different control mechanisms. The first scenario involved the continuous control of musical parameters by using the location of the user's body to select the instrument, and their hand positions to continuously control musical parameters. The second implemented restricted parameter control by using discrete gestures. The third scenario used gestures to select an instrument and pose landmark positions to affect the musical parameters of that instrument. Table 5.1 shows a summary of the control mechanisms used in each scenario.

This section opens a discussion on the user experience and the impact that the computational model had on the user experience. Throughout the user study, it became apparent that the profile of users significantly impacted the experience of the three scenarios. For this reason, the following discussion will present three user profiles and their intention to differentiate between the control scenarios described in Section 3.4, in the form of users with less-experience in movement-based practices such as dance, yoga and martial-arts, experienced movers, and users with the intent of using the instrument as a compositional tool.

Table 5.1: Summary of the instrument activation and parameter control techniques of the three scenarios used for the study.

Scenario	Instrument Activation	Parameter Control	Control Type	Instruments
1	Body location	Wrist location	Continuous	• Bubble • Harp
2	Body location	Gesture	Discrete	• Bubble • Harp
3	Gesture recognition	• Wrist location • Arm angle	Continuous	• Bubble • Harp • Wave

5.1 Human Interface

The results from the user study suggest that the ability to continuously control parameters (Scenario 1 and 3) is a considerably more favourable method than using gestures to control discrete parameters (Scenario 2). Discrepancies between the enjoyment and intuition of scenario 1 and scenario 3 can be analysed by assessing the profile and intention of the user.

Users Less Experienced in Movement-Based Practices

In scenario 1, participants with less experience in movement-based practices often took more time to work out the instrument, and moved their hands with more constrained motion to control the pitch and tempo. Scenario 3 was the most enjoyable as it stimulated greater movement through the activation of certain instruments using gestures and with users describing it as a full body experience. This relationship between an instrument and a gesture was more intuitive for some participants, enabling a more immersive experience. One participant said “the movements in scenario 3 linked to one another more and therefore so did the sounds. I enjoyed scenario 1 and 2, however I was more actively thinking about how I controlled sounds.” Such comments suggest that there is greater freedom afforded when using gestures to identify instruments as instruments can have an associated movement. One user who had suffered a stroke in the last year said “it got me to move in a different way.” This method of activating certain sounds using discrete gestures could be useful if applied to a public interaction or clinical use, in which encouraging greater dynamism or specific movements are desirable.

Users Highly Experienced in Movement-Based Practices

Participants with a high level of experience in movement-based practices used the instrument with extreme fluidity. In these cases, it was surprising the speed at which the participants appeared to grasp the instruments, suggesting perhaps the ease-of-use for those who are confident when using their bodies in dynamic ways, irrespective of the control mechanism. This was reflected in the results, as both these users rated all three scenarios 10 for intuition. The use of hand locations in scenario 1 arguably facilitated greater freedom for which the instrument parameter space could be explored, compared to exploring an instrument within the physical constraints of the gesture that activated it (scenario 3), which is dependent on the design of the gesture recognition algorithm. Table 5.2 shows a comparison between the continuous parameter control methods when using gestures (Scenario 3) compared to hand location (Scenario 1).

As a Composition Tool

Another observation came when evaluating the experience had by musicians and those who intended to compose music. One user, a professional musician, commented that "Scenario 1 had the most potential". Indeed, the ability to carefully control the sound of an instrument is a crucial element for any musician. It should be no coincidence, therefore, that the control mechanisms of interactive digital musical instruments (DMIs) typically utilise hands compared to other body limbs, as they are more naturally intuitive control mechanisms. This reinforces the notion that when used as a compositional device or where the amount of control is imperative, the use of hands may be the most suitable. However, as an exploratory device used to stimulate movement, full body gestures can be used to great effect, and can serve a more enjoyable and unique experience.

One advantage of using gestures to activate instruments (scenario 3) compared to location (scenario 1 and 2), was that it became possible to layer instruments, therefore creating more textured soundscapes. This worked particularly well in scenario 3 when one could overlap the sounds of sudden gestures (i.e., the kick and punch corresponding to triggering waves) onto the sounds mapped to sustained gestures (e.g., floating corresponding to the harp sound). One user described the ability to merge multiple sounds felt "more like making music versus playing an instrument." Nonetheless, there were still some limitations of the design as a compositional tool. One participant said they would "like to loop the sounds", and another said "it would be interesting for the

program to understand my rhythm." These features could be achieved with use of a more sophisticated machine learning model that recognises temporal features, or one that enabled greater parameter control such as the use of hand gestures.

Table 5.2: The advantages and disadvantages of continuous control mechanisms that utilise full body gestures compared to hand location and their potential applications.

Control Mechanism	Advantages	Disadvantages	Applications
Full-body gesture	• Association of gesture to instrument can be intuitive for users with less movement experience • Stimulate specific movements • Greater movement stimulated • Ability to layer instruments	• Limited by design of gesture-recognition algorithm	• Public • Medical
Hand location	• Greater parameter control • Less constrained by having to perform specific gestures • More reliable • Ability to layer instruments	• Movement-parameter control less intuitive for users with less musical experience • Less movement stimulated for less expereinced movers	• Composition • Dance

5.2 Impact of Computational Model on User Experience

5.2.1 Data Collection

The mapping of discrete gestures to control pitch and tempo in scenario 2 was less successful. One user made the observation that "the punch gesture struggled to register." This problem occurred repeatedly throughout the user study, highlighting a fundamental issue of collecting appropriate and accurate data. In the training phase, users were required to be in the gestural position at the end of a five second timer. However, for the punch gesture, many samples were captured of a person when moving into a punch rather than already being in the punch position at the point of capture. This was not a problem for flick and float gestures as they were sustained movements,

and so were less varied throughout the training process.

Another observation was the regular triggering of the wave sound in scenario 3 due to false positive classification of the kick gesture. When collecting training samples for the neutral gesture, users were asked to walk around the performance area. However, the labelling of neutral pose samples did not represent the variation of neutral poses that were performed in the user study. For example, when participants crouched and moved into another gesture, due to a crouching position not being represented in the training data, the separation of the user's legs often resulted in the recognition of a kick gesture. As shown by Figure 5.1, the probability of the kick gesture was particularly high when in a neutral position.

Another limitation arose in scenarios 2 and 3, when an elderly participant attempted to activate control parameters and instruments from a seated position. The training data did not include gesture samples from a seated position. For this reason, the algorithm failed to correctly classify the user's intended gesture, and therefore did not function as intended.

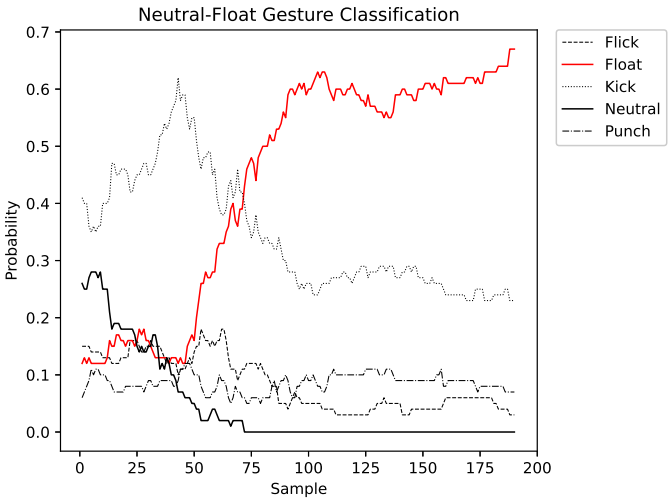


Figure 5.1: Gesture probability moving from neutral to float gestures illustrates limitation of gesture-classification algorithm with high kick gesture probability as arms are raised into floating gesture prior to its detection at approximately 75 samples.

5.2.2 Pose Estimation Model

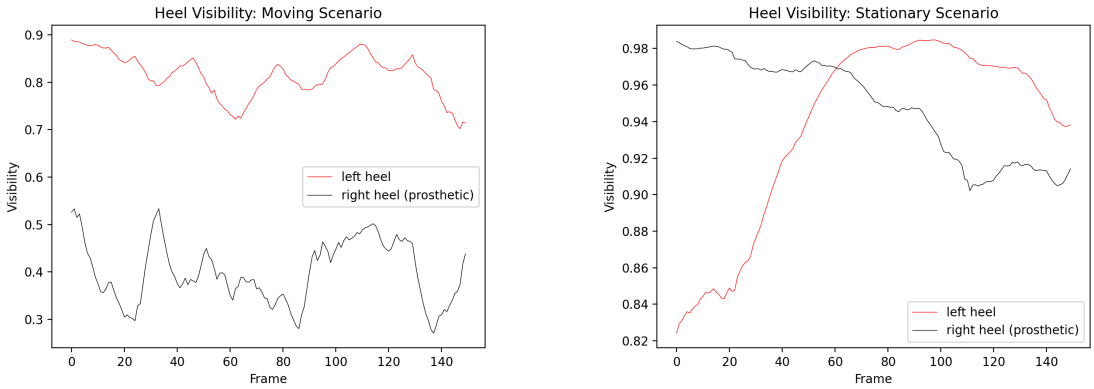
Limitations of the pose estimation model, which did not infer landmarks on the hands, meant that hand gestures could not be differentiated. This resulted in punches (in which the non-punching hand was also in front of the waist) being regularly misclassified as flicking gestures.

Robustness to Prosthetic Limbs

This model has been primarily used on people with typical bodies. Disabled communities are often marginalised from music participation due to physical disability. To explore the suitability of the pose estimation model among disabled users, the robustness of the model was evaluated on a video of a man with a prosthetic leg walking (i.e. continuous movement). A second scenario showed the same man stretching, with his standing prosthetic leg stationary (snapshot of pose landmark inference shown in Appendix B). Under both scenarios, the visibility confidence of the prosthetic heel was significantly lower than the non-prosthetic heel. Figure 5.2a shows that the prosthetic heel visibility is greatly reduced when undergoing continuous movement (0.287-0.541) compared to that of the non-prosthetic heel (0.691-0.894). The visibility of the stationary prosthetic heel (Figure 5.2b) is significantly higher than when undergoing continuous motion. If the system were applied for people with prosthetic limbs or without limbs, the lower confidence value of the pose detection model suggests that further work, such as the optimisation of pose model parameters, would be required prior to deployment. However, fundamental limitations of the pose estimation model dataset due to fewer labelled training instances of people with prosthetic limbs or without certain limbs may hinder the functionality of the pose estimation model when used among physically disabled communities.

5.2.3 Sound Module

The broad user response to each instrument was that the bubble generator was easier to control, but the harp sound gave greater depth and variability. This was a direct consequence of the the granular synthesis module as the use of a small sample was likely to yield more intuitive control, but with potentially less depth and unpredictability. One user remarked “once I understood the movement [of the harp sound] it was most enjoyable to manipulate and could make more varied noises.” The samples used for an interactive granular synthesiser should therefore be considered depending on the application and intended function.



(a) (b)

Figure 5.2: Graph comparing visibility of heel landmark on prosthetic and non-prosthetic leg for two scenarios using MediaPipe Pose model (a) continuously moving prosthetic leg, (b) stationary prosthetic leg.

The fact that users enjoyed the unpredictability of the sound despite them not having full control on its output, shows that generative sound is successful when used in a context of human-machine interface for sound generation. Moreover, it demonstrates that parameterisation of a complex process in granular synthesis to user-friendly control parameters in pitch and tempo is a good strategy for the employment of generative human-machine sound interaction. One user commented that "the quality of sounds made the experience pleasurable and relaxing" whilst another said they "could have played [the instruments] in a meditative way for ages." The use of audio in rehabilitative studies have typically used pre-composed music or speech [8]. Results from this study indicate that the use of generative music in clinical situations may be well suited.

5.3 Conclusion

Findings from this report suggest that one is better placed to understand the opportunities and challenges that come with designing an audio feedback system that utilises movement-based control inputs. The exploration of multiple movement-based control mechanisms for a generative audio feedback system was successfully achieved. The system employed human pose estimation to fulfil accurate real-time single-person pose detection at 20 fps, a gesture classification algorithm and a generative music system using a granular synthesis sound module.

Three control scenarios were tested to evaluate the user experience of continuous or restricted instrument parameter control when using hand positions or discrete gesture as control mechanisms. All participants in a user study favoured continuous parameter control compared to discrete control. For continuous parameter control, the use of hand location (scenario 1) was the most intuitive control mechanism for users with significant experience in movement-based practices, and this afforded greater flexibility than the use of gestures which is more dependent on the design of the gesture classification algorithm. Applications that may utilise hand locations may include dance. For less naturally-dynamic movers, the association of mapping gestures to an instrument (scenario 3) was more intuitive and stimulated greater dynamism. This also demonstrates the success of applying Laban Movement Analysis to map movement to music. This form of control may be well suited to public-use and medical applications, where stimulating specific movements and engagement is desired. The mapping of gestures to discrete control parameters (scenario 2) was largely unsuccessful, with participants finding less relation between the control gesture and its associated control parameter. It should, however, be acknowledged that despite the 98.6% accuracy of the Random Forest Classification algorithm itself, erroneous data collection impacted the user enjoyment when used for discrete parameter control as certain gestures were not recognised as intended.

The project successfully achieved its aim in designing an enjoyable sounding feedback system. The unpredictable sound was enjoyed by 89% of users, illustrating the successful deployment of generative music and granular synthesis for music interaction at a low-skill entry point. The aim to design a modular system for which new instruments could be added was partially successful, however, given significant design parameterisation of the granular synthesis module to obtain user-friendly audio output further work is required to achieve a fully adaptable module for which new instruments can be realised without significant design intervention.

5.4 Further Works

The extension of the pose estimation model to also utilise hand landmarks (e.g. MediaPipe Hand), would also enable greater parameter control, and could enhance the instrument for compositional purposes, such as looping. Moreover, the use of hand tracking would enable greater distinction between gestures, such as punching (closed fist) compared to pushing (open hand), increasing the classification accuracy of the

system.

If self-capturing data, a larger training user group should be used and the process of collecting the data should be clarified to the user, so as to avoid any confusion that may lead to false positive classifications, and also represent a large variation within each class. A robust data collection method could be to implement transfer learning to classify the desired gestures within a pre-trained model of thousands of labelled images of people. However, transfer learning is limited in its lack of flexibility, as certain gestures may be infrequent or non-existent in the dataset of the pre-trained model used, and therefore may not represent the variation required for accurate classification on unseen data. In order for the system to be applicable to the disabled community, representative training data must be collected.

Developments to detect multiple people for group music interaction would be interesting, however, the deployment of multi-pose recognition systems is significantly more complex and would require considerably greater hardware. Such a system would likely require a GPU or neural accelerator. The implementation of a model with the ability to map temporal features such as Dynamic Time Warping or Hidden Markov Model would enable the potential for a more developed parameter control space, such as a greater distinction between temporal aspects of Laban movements (e.g., sudden or sustained movements), or the ability to recognise patterns in the movement of a user.

References

[1] ABE, K., TAKANE, S., and SATO, S. (2015). Investigation on influence of additional sound on comfortableness of living environment. *Interdisciplinary Information Sciences*, 21(2):151–157.

[2] Andriluka, M., Pishchulin, L., Gehler, P., and Schiele, B. (2014). 2d human pose estimation: New benchmark and state of the art analysis. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3686–3693.

[3] Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., and Grundmann, M. (2020). Blazepose: On-device real-time body pose tracking. *arXiv preprint arXiv:2006.10204*.

[4] Beets, I. A., Macé, M., Meesen, R. L., Cuypers, K., Levin, O., and Swinnen, S. P. (2012). Active versus passive training of a complex bimanual task: is prescriptive proprioceptive information sufficient for inducing motor learning? *PLoS One*, 7(5):e37687.

[5] Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., and Black, M. J. (2016). Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *European conference on computer vision*, pages 561–578. Springer.

[6] Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., and Sheikh, Y. (2021). Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1):172–186.

[7] Caramiaux, B., Montecchio, N., Tanaka, A., and Bevilacqua, F. (2014). Adaptive gesture recognition with variation estimation for interactive systems. *ACM Trans. Interact. Intell. Syst.*, 4(4).

[8] Cavalcanti, V. C., de Santana Ferreira, M. I., Teichrieb, V., Barioni, R. R., Correia, W. F. M., and Da Gama, A. E. F. (2019). Usability and effects of text, image and audio feedback on exercise correction during augmented reality based motor rehabilitation. *Computers & Graphics*, 85:100–110.

[9] Dieleman, S., van den Oord, A., and Simonyan, K. (2018). The challenge of realistic music generation: modelling raw audio at scale. *Advances in Neural Information Processing Systems*, 31.

[10] Dillon, S., Adkins, B., Brown, A., and Hirche, K. (2009). Communities of sound: Examining meaningful engagement with generative music making and virtual ensembles. *International Journal of Community Music*, 1(3):357–374.

[11] Dipietro, L., Sabatini, A. M., and Dario, P. (2008). A survey of glove-based systems and their applications. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(4):461–482.

[12] Donahue, C., McAuley, J., and Puckette, M. (2018). Adversarial audio synthesis. *arXiv preprint arXiv:1802.04208*.

[13] Dong, J., Jiang, W., Huang, Q., Bao, H., and Zhou, X. (2019). Fast and robust multi-person 3d pose estimation from multiple views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7792–7801.

[14] Engel, J., Agrawal, K. K., Chen, S., Gulrajani, I., Donahue, C., and Roberts, A. (2019). GANSynth: Adversarial neural audio synthesis. In *International Conference on Learning Representations*.

[15] Fails, J. A. and Olsen, D. R. (2003). Interactive machine learning. In *Proceedings of the 8th International Conference on Intelligent User Interfaces*, IUI '03, page 39–45, New York, NY, USA. Association for Computing Machinery.

[16] Fang, B., Sun, F., Liu, H., and Liu, C. (2018). 3d human gesture capturing and recognition by the immu-based data glove. *Neurocomputing*, 277:198–207.

[17] Fang, H.-S., Xie, S., Tai, Y.-W., and Lu, C. (2017). Rmpe: Regional multi-person pose estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 2334–2343.

[18] Fiebrink, R. and Caramiaux, B. (2016). The machine learning algorithm as creative musical tool. *arXiv preprint arXiv:1611.00379*.

[19] Fiebrink, R., Trueman, D., Britt, N. C., Nagai, M., Kaczmarek, K., Early, M., Daniel, M., Hege, A., and Cook, P. R. (2010). Toward understanding human-computer interaction in composing the instrument. In *ICMC*. Citeseer.

[20] Finocchietti, S., Cappagli, G., Porquis, L. B., Baud-Bovy, G., Cocchi, E., and Gori, M. (2015). Evaluation of the audio bracelet for blind interaction for improving mobility and spatial cognition in early blind children - a pilot study. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 7998–8001.

[21] Gałka, J., Mąsior, M., Zaborski, M., and Barczewska, K. (2016). Inertial motion sensing glove for sign language gesture acquisition and recognition. *IEEE Sensors Journal*, 16(16):6310–6316.

[22] Gioti, A.-M. (2019). Imitation game: Real-time decision-making in an inter-active composition for human and robotic percussionist.

[23] Gioti, A.-M. (2021). Artificial intelligence for music composition. In *Handbook of Artificial Intelligence for Music*, pages 53–73. Springer.

[24] Guo, L., Lu, Z., and Yao, L. (2021). Human-machine interaction sensing technology based on hand gesture recognition: A review. *IEEE Transactions on Human-Machine Systems*.

[25] Imogen and Heap (2014). Mimu gloves.

[26] Jiang, N., Falla, D., d'Avella, A., Graimann, B., and Farina, D. (2010). Myoelectric control in neurorehabilitation. *Critical Reviews™ in Biomedical Engineering*, 38(4).

[27] Kiefer, C., Collins, N., and Fitzpatrick, G. (2009). Phalanger: Controlling music software with hand movement using a computer vision and machine learning approach.

[28] Laban, R. and Ullmann, L. (1971). The mastery of movement.

[29] Li, H., Xie, H., and Xin, S. (2017). A soundwalk study with eeg test in the mountainous urban park. In *24th International Congress on Sound and Vibration, ICSV*, volume 2017.

[30] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

[31] Mellace, S., Guzzi, J., Giusti, A., and Gambardella, L. M. (2019). Realtime generation of audible textures inspired by a video stream. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9865–9866.

[32] Moreno-Noguer, F. (2017). 3d human pose estimation from a single image via distance matrix regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[33] Motion, L. (2015). Leap motion. *San Francisco, CA, USA*.

[34] Mroz, S., Baddour, N., McGuirk, C., Juneau, P., Tu, A., Cheung, K., and Lemaire, E. (2021). Comparing the quality of human pose estimation with blazepose or openpose. In *2021 4th International Conference on Bio-Engineering for Smart Technologies (BioSMART)*, pages 1–4. IEEE.

[35] Munea, T. L., Jembre, Y. Z., Weldegebriel, H. T., Chen, L., Huang, C., and Yang, C. (2020). The progress of human pose estimation: A survey and taxonomy of models applied in 2d human pose estimation. *IEEE Access*, 8:133330–133348.

[36] Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

[37] Papandreou, G., Zhu, T., Kanazawa, N., Toshev, A., Tompson, J., Bregler, C., and Murphy, K. (2017). Towards accurate multi-person pose estimation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4903–4911.

[38] Riva, G., Baños, R. M., Botella, C., Mantovani, F., and Gaggioli, A. (2016). Transforming experience: the potential of augmented reality and virtual reality for enhancing personal and clinical change. *Frontiers in psychiatry*, 7:164.

[39] Roads, C. (2004). *Microsound*. The MIT Press.

[40] Sapp, B. and Taskar, B. (2013). Modec: Multimodal decomposable models for human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3681.

[41] Sathiyanarayanan, M. and Rajan, S. (2016). Myo armband for physiotherapy healthcare: A case study using gesture recognition application. In *2016 8th International Conference on Communication Systems and Networks (COMSNETS)*, pages 1–6.

[42] Schaffert, N. and Mattes, K. (2015). Interactive sonification in rowing: Acoustic feedback for on-water training. *IEEE MultiMedia*, 22(1):58–67.

[43] Schramm, R., Jung, C. R., and Miranda, E. R. (2015). Dynamic time warping for music conducting gestures evaluation. *IEEE Transactions on Multimedia*, 17(2):243–255.

[44] Sun, K., Xiao, B., Liu, D., and Wang, J. (2019). Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5693–5703.

[45] Tahiroğlu, K., Svedström, T., and Wikström, V. (2015). Noisa: A novel intelligent system facilitating smart interaction. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, page 279–282, New York, NY, USA. Association for Computing Machinery.

[46] Tanaka, A., Di Donato, B., Zbyszynski, M., and Roks, G. (2019). Designing gestures for continuous sonic interaction.

[47] Toshev, A. and Szegedy, C. (2014). Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660.

[48] Uhlenbeck, G. E. and Ornstein, L. S. (1930). On the theory of the brownian motion. *Physical review*, 36(5):823.

[49] Van Nort, D. (2018). Genetically sonified organisms: Environmental listening/sounding agents. In *Proceedings of the Musical Metacreation Workshop*.

[50] Viglialoro, R. M., Condino, S., Turini, G., Carbone, M., Ferrari, V., and Gesi, M. (2019). Review of the augmented reality systems for shoulder rehabilitation. *Information*, 10(5):154.

[51] Visi, F. G. and Tanaka, A. (2020). Towards assisted interactive machine learning: exploring gesture-sound mappings using reinforcement learning. In *ICLI 2020—the fifth international conference on live interfaces*, pages 9–11.

[52] Yamamoto, R., Song, E., and Kim, J.-M. (2020). Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6199–6203.

[53] Yang, L.-C., Chou, S.-Y., and Yang, Y.-H. (2017). Midinet: A convolutional generative adversarial network for symbolic-domain music generation. *arXiv preprint arXiv:1703.10847*.

[54] Zhang, Y., An, L., Yu, T., Li, X., Li, K., and Liu, Y. (2020). 4d association graph for realtime multi-person motion capture using multiple video cameras. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1324–1333.

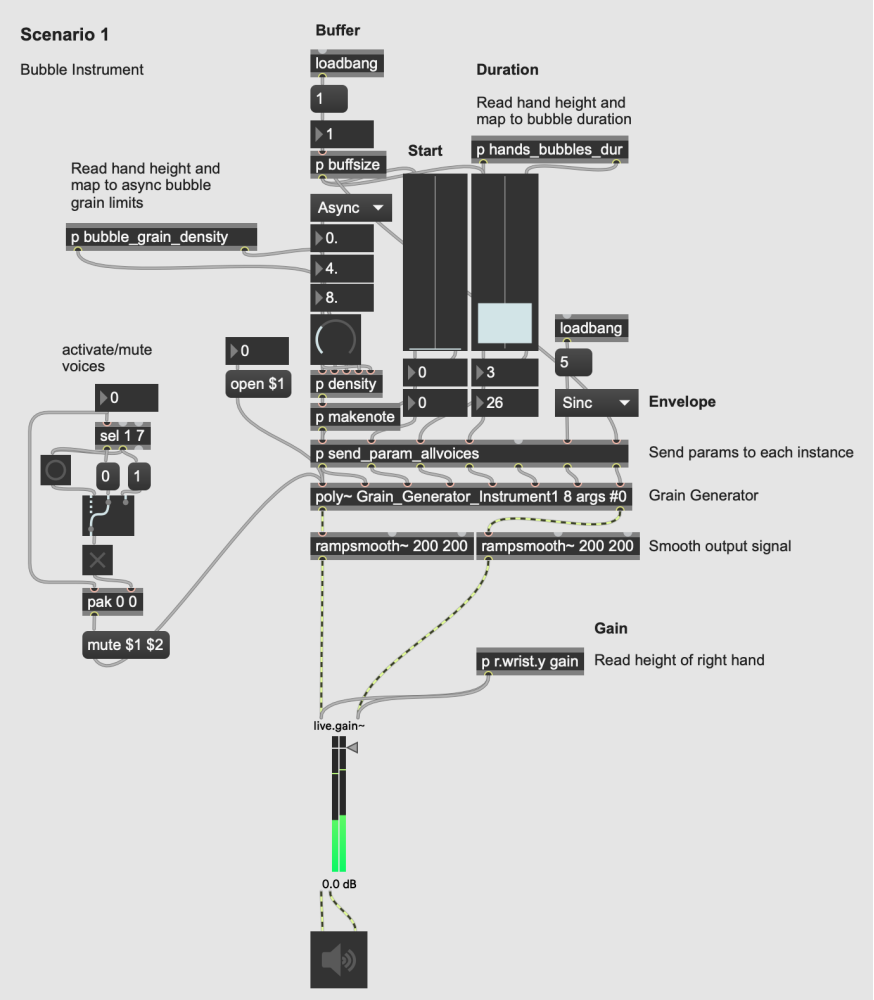
[55] Zhang, Z. (2012). Microsoft kinect sensor and its effect. *IEEE multimedia*, 19(2):4–10.

[56] Zhou, X., Zhu, M., Leonardos, S., Derpanis, K. G., and Daniilidis, K. (2016). Sparseness meets deepness: 3d human pose estimation from monocular video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4966–4975.

Appendix A

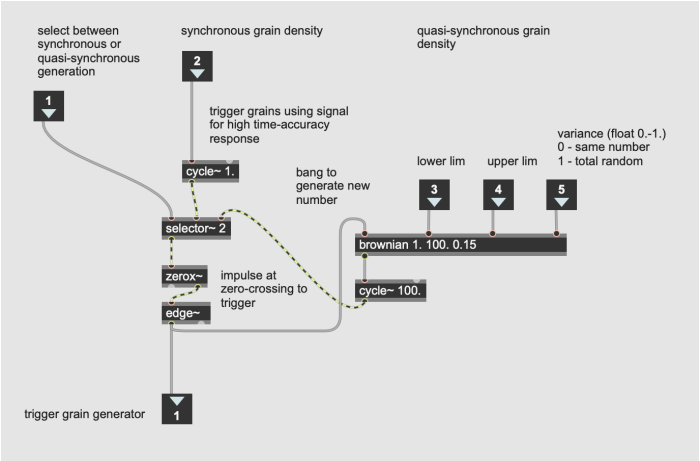
Granular Synthesis Module

Granular synthesis instrument used for bubble instrument in scenario 1 that uses hand landmarks to control granular synthesis parameters:



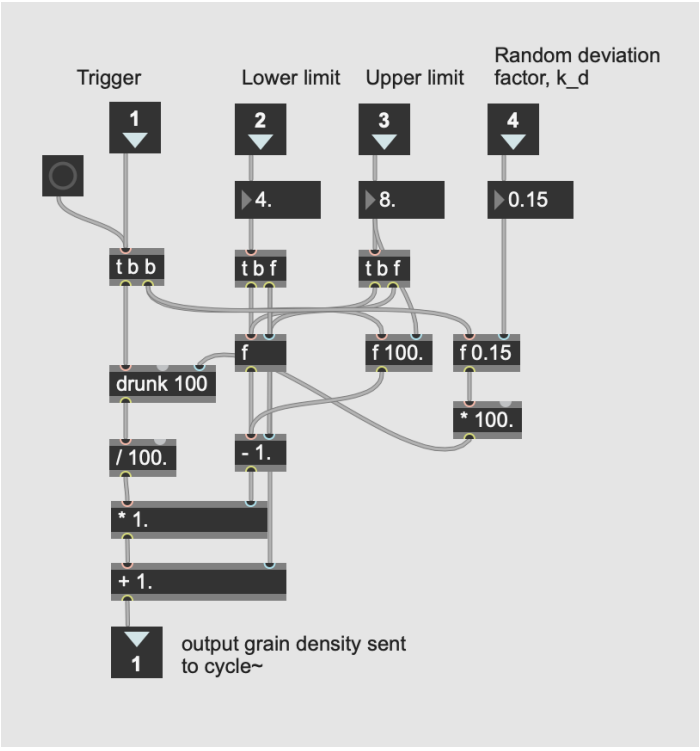
Grain Density

Patch that selects type of grain triggering and by triggers grain generator at impulse:



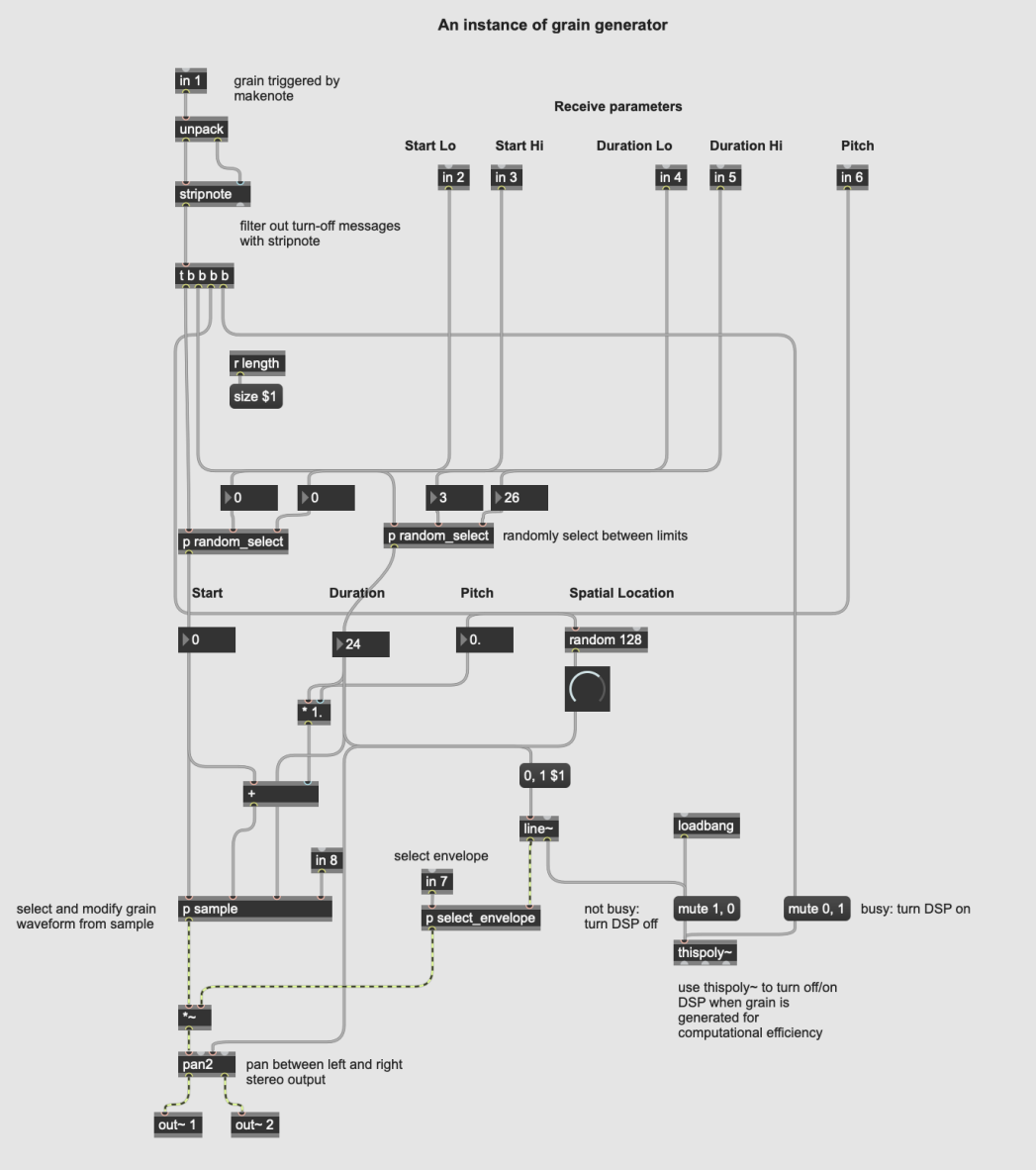
Brownian Motion

Patch to generate random Brownian number:



Grain Generator

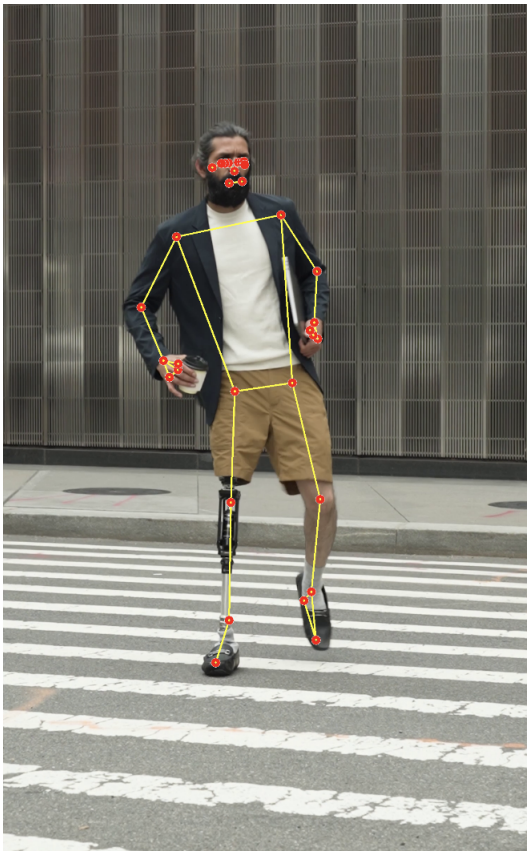
An instance of the grain generator:



Appendix B

Pose Estimation

Snapshot of pose detection on man with prosthetic leg



(a)

(b)