Dance2Vec: Encoding Movement Qualities from 2D Video Footage

William Primett

333467905
MSci Creative Computing
Department of Computing
Goldsmiths, University of London

May 2018

ABSTRACT

Human movement qualities great provide a potential for accommodating rich interactive experiences, given the profound traits that provide these physical gestures with so much expressive scope. Where some qualities cannot be directly hard- coded, a machine learning approach is often proposed to substantiate a semantic meaning towards digital representations of human movement. In this project, we introduce a range of methods for extracting biometric and visual features from a corpus of video segments, which are then embedded into a continuous vector space according to the qualities of movement present in the source footage. The complete workflow contains multiple tiers of dimensionality reduction and feature extraction that resolves towards a 2D representation. To test and compare different configurations of our model, we construct a test that asks participants to rate the perceptual similarity of segments matched by the system against those selected at random. By evaluating the responses, we gain insight on how well our feature set aligns with our perception of human movement and how it could be effective towards other computational models.

KEYWORDS

Movement Qualities, Dimensionality Reduction, Machine Learning, Post Estimation

Table of Contents

Table of Figures

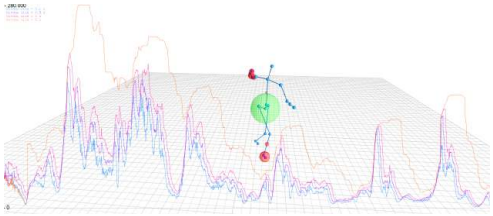

Figure 1 MotionMachine Screenshot

Visualization of calculated Weight Effort [20]

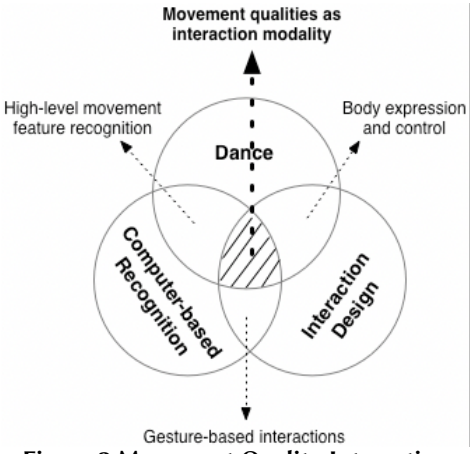

Figure 2 Movement Quality Interaction Framework

Taken from Movement Qualities as Interaction Modality [16]

1 INTRODCTION

1.1 Report Structure

This report is organised as follows: We start by summarizing some key concepts and related research topics which will support the desired outcomes for the project. Then we describe the tests that are later used to evaluate this approach. The design process will run through the decisions made during development before explaining the technical details of the final workflow. Finally, I'll go onto discussing the test results and review the overall validity of the system.

1.2 Background Research & Motivation

Before introducing the details of the project, we'll review the following literature that influenced our overall approach.

1.2.1 Movement Qualities as Interaction Modality (2012)

This paper hypothesizes the use of movement qualities in interactive systems as a means to promote physical exploration and expression. A user experiment is conducted with the installation, A light touch. Participants learn and use movement qualities to control the position of a spotlight. The evaluation of user feedback indicates the user's experience whilst using movement quality against traditional position-based interactions in the same setting. The outcome showed users favouring movement qualities in aspects of "richness" and were rated more "intriguing". They were also observed spending more time exploring different interactions. One the other hand, users found the movement qualities harder to learn and perform, rated less 'intuitive' and 'natural' as a result. [16]

1.2.2 Seeing, Sensing and Recognizing Laban Movement Qualities (2017)

This study investigates effective practices for designing computational model for recognizing LMA Effort qualities. It concludes that:

- Computational modelling of Effort qualities should be informed by the expertise of Certified Movement Analysts (CMAs)
- Appropriate Multimodal data should be utilized to gain a wider scope for movement quality recognition from a richer set of low-level data. In this study, they combine: positional (motion capture), dynamic (Accelerometers) and physiological (EMG) data to represent Space, Time and Weight.
- Where video data was a limited resource for CMA's to identify Space Efforts, they suggested using Shape qualities to describe the mover's spacial attention. Correlating Direct and Indirect Efforts with Enclosing and Spreading actions respectively.

1.2.3 Semantic Segmentation of Motion Capture Using Laban Movement Analysis (2007)

This report presents a method for retrieving meaningful snippets of data from large streams of motion capture data. An LMA classifier is constructed from a set of four neural networks to represent the continuum of each Effort component. The authors explain the potential applications for automated segmentation and labelling of low-level motion data.

1.2.4 Motion Data and Machine Learning: Prototyping and Evaluation (2015)

The MotionMachine software presented in this report provides an efficient workflow for visualizing and annotating motion data, made suitable to train and validate machine learning models with user-selected segments of data. The report justifies the importance of such tools when applying human-in-the-loop machine learning to motion data, where users are can rapidly assemble and test models **Error! Reference source not found.**

1.2.5 Sketches vs Skeletons: video annotation can capture what motion capture cannot (2015)

This study sets out to evaluate the use of motion capture technologies to assess one's posture during musical performance. However, during the initial user feedback, they discover this approach to be unreliable due to the glitches occurring in the data. The authors go on to explain the importance of the contextual cues that are abstracted from skeletal data. As a result, a fundamental redesign is proposed and developed, using computer-vision based motion tracking to aid users annotate video footage.

1.2.6 Layering the Choreographic Process: Making Dance Work with Machine Learning (2017)

Following their work on combining algorithms with real-time choreographic processes, Dr. Kate Sicchio presents a new performance that's influenced by machine learning algorithms. In "*Untitled Algorithmic Dance 2*", a performer is captured during an improvised sequence accumulating 3000 images. The images are clustered and visualised along using the t-SNE dimensionality reduction algorithm which in turn generates a new choreographic score. It's concluded that this method was capable of pushing one's choreographic possibilities, recognizing its use as an effective human centred tool. The report finishes with perspectives to follow up this approach with motion capture initiatives [6].
This practice work inspires the application for clustering video data, which represents continuous motion over static poses.

The project takes its name from the highly popular Word2Vec model[15]. In their paper, Mikolov et al. explain how they are able to distribute words in a continuous vector space according to their semantic relationship to one another; this makes it data suitable for statistical analysis. Furthermore, we are able to visualize these relationships with the use of dimensionality reduction algorithms, observing how words with similar meanings are 'clustered' together. A similar approach can be applied to audio and image data by extracting the perceptually important features that are already encoding in the data.

Previous studies report the use of machine learning techniques for deriving semantic qualities from low-level, high dimensionality motion data. Normal two-dimensional video provides a highly accessible modality for recording, sharing and annotating examples of human movement. However, its pixel-based representation is highly inefficient as an input for a computational model. For Dance2Vec, we exploit the most recent state-of-the-art model for pose estimation to derive features of movement from 2D video footage. If we were able to represent these features in a low-dimensionality representation that can be visualized and linked back to their source, how might this influence the contemporary approach to tasks involving machine learning and human movement data?

1.3 Approach Overview

To outline the key elements of our workflow (illustrated in **Figure 2**), we separate our system into the following steps, which are described in more detail throughout the report.

- In order to derive skeletal information from a standard 2D video, we used the open source OpenPose system **[5]**. OpenPose is a deep learning framework for keypoint body estimation **[7]** that uses the COCO common objects human pose dataset **[2]** to estimate *x* and y coordinates for each frame where a human figure is detected. In our workflow, the output from OpenPose feeds the feature extraction through a JSON stream (one file per video frame).

- By taking keypoint positions for each frame, we extract a set of normalized biometric features for the entire video clip that is then exported as one file. Global features, that encompass the cumulative dynamics of all the joints, are uses to divide the original clip into sub-segments that each display a specific movement, or what's later described as a '*period of kinematic exertion'* in section 2.3. These are the segments that will be used for our final video embeddings.

- For each segment, we acquire two types of features. One we'll refer to biometric, concerning the subjects relative position of the subject's joints; these are derived from the exported feature file which is split according to the frames occupied by each segment. The other set of features account for the way the subject's movements appear on-screen, using computer vision techniques to analyse each frame, merging the results into a single image. We'll refer to these as our visual features.

- From here, we concatenate these features into single vectors that comprise our full dataset; this is used as an input towards three dimensionality reduction algorithms: t-SNE, UMAP and PCA. Once the segments are assigned a position, they are plotted in an online interactive visualizer.

Interactive demo: http://igor.gold.ac.uk/~wprim001/MOCO/demo/demo.html
(Recommended Configuration: Raw Video, t-SNE, 4/5 Perplexity, 2/5 Iterations)

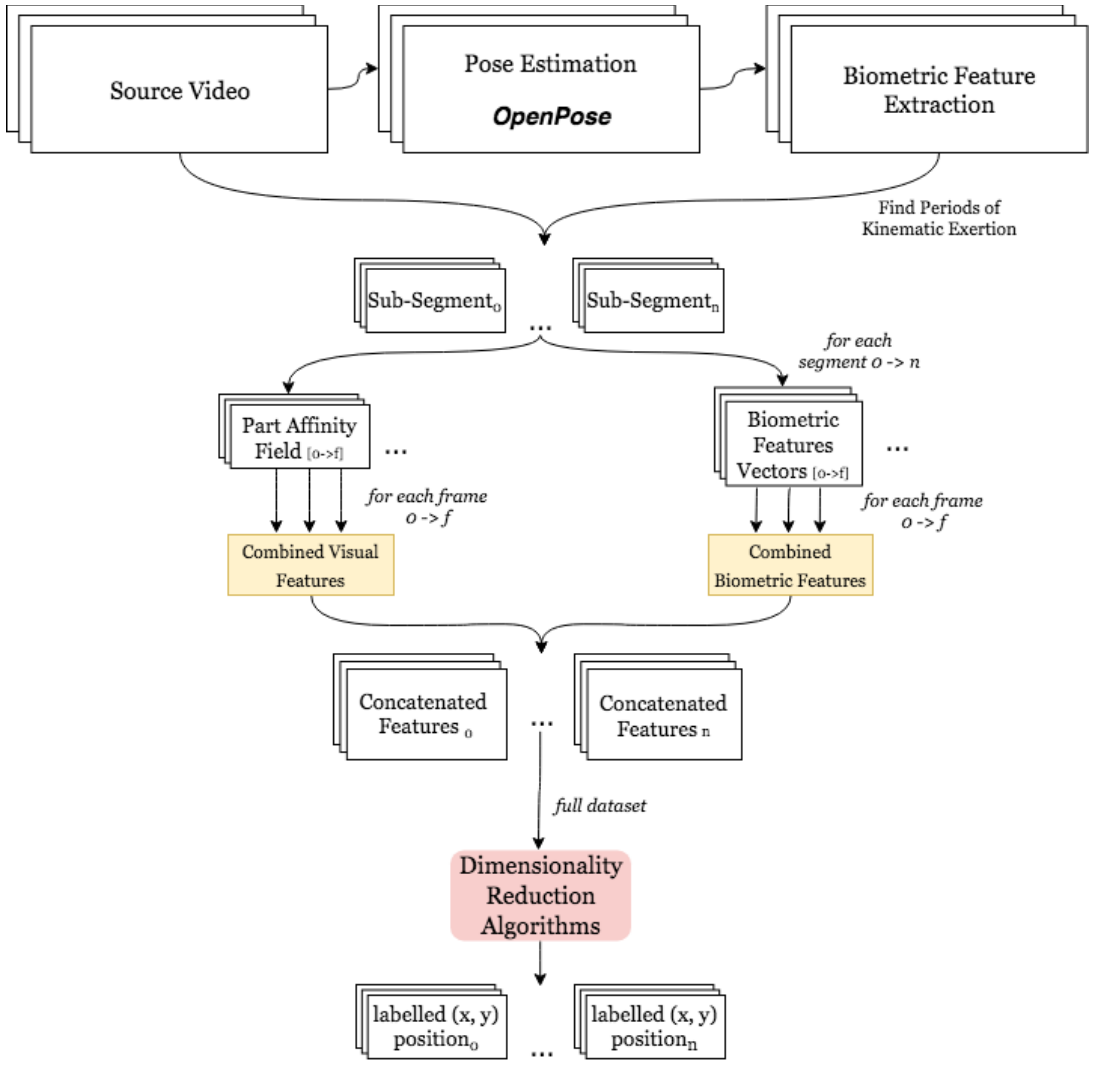

Figure 2 Dance2Vec Workflow

1.4 Project Aims

- We intend to reliably recognize movement qualities from video footage regardless of the performer or environment, extracting features that solely represent motion. Calculations should be normalized in accordance to the size of the video and the relative scale of the performer.
- Where possible, we'd like to be able to identify demonstrations of individual LMA Efforts Elements when explicitly demonstrated. In Addition, we'd like to detect different combinations of Efforts components that constitute an observable Basic Effort Action (BEA) [3].
- The system should be able to match segments where the same choreography is being performed by the subject. A selection of our data contains the same composition of movements performed by different dancers or during a different performance. When a consecutive stream of segments is fed through the system, corresponding segments should be matched accordingly so the output closely mimics the input.
- Where near-identical segments aren't available in the dataset, we should observe suitable pairings that capture some of the expressive characteristics of the input. In this case, the parallels may hold looser descriptions, such as those referring to the general shape or direction of the subject's movement

1.5 Datasets

Human movement qualities are apparent in a vast range of physical disciplines, the focus on dance in this study can be justified for the following reasons:

- An extensive amount of high-quality video footage of dancers is widely available from the public domain, where all joints are usually visible towards the camera, making it compatible with the OpenPose framework (see section 2.1)
- Dance is proficient in demonstrating a dynamic fluctuation of expressive gestures within a single sequence, allowing for intuitive transitions between physical states
- Where generic human action datasets often comprise examples of functional movements, our focus towards dance practice sets our attention towards expressive movement qualities
- In a performance context, the expressive qualities of gestures can be associated with those formalized in the domain of music. For example, LMA Effort is often compared to dynamic musical features such as *legato, forte, dolce, etc…[18]* which inform the emotional content of the performance

Our video corpus is derived from three main sources:

1. For the LMA Effort ground truth, we use a publicly available repository of labelled movement videos from Eckert's *"Genetic Dance Algorithm"* project [4]. The videos are frontal studio recordings of simple limb, head, and torso movements that are categorised according to LMA effort. They were recorded in two separate sessions, one black and white, and another on green screen. We used footage from the two sessions, adopted as training and test data for our preliminary classification task in 0.
2. We'll also test our approach with video clips from *Merce Cunningham's Split Sides*, divided into a continuous stream of shorter segments that comprise a sub-sequence of solo choreography that is performed on two occasions to separate soundtracks. The aim is to have the individual segments matched to where the choreography is repeated, along with other segments that contain perceptually similar movements [10]. It is also worth noting that the examples of movement demonstrated in these clips are considered highly complex and are less confirmative to any traditional vocabulary for movement analysis.

Throughout his career, Cunningham was considered a pioneer of utilizing new technologies within the choreographic practice. During the 1970's, he highly endorsed the use of film to record dance. He'd later go on to pioneer the use of the DanceForms/LifeForms software as well as integrate motion capture to create visuals for his performances. [19]

His passion for exploration and innovation would take his work beyond the boundaries of traditional dance notation

"Don't be surprised, if you find it difficult to find words to describe those movements. The situation is very much like it was when the musical vocabulary had to free itself from Italian. The ballet vocabulary comes from a time when the torso wasn't used at all. For example, when Margaret Jenkins wrote down Summerspace in the Laban notation, certain things in the dance weren't there because they weren't in the notation" ***Merce Cunningham, The Dancer and the Dance p. 119 [17]***

Figure 3 Datasets

Top: *Genetic Dance Algorithm* [4]

Bottom-left: *Prix de Lausanne* [8]

Bottom-right: Merce Cunningham Dance Company: Split Sides [10]

3. To complete our corpus with a substantial verity of movement data from a diverse range of performers, we acquire data from the *Prix de Lausanne* online media collection. *"The Prix de Lausanne, an international competition for young dancers, is open to young dancers of all nationalities aged 14 to 19 who are not yet professionals."* [8]
   The annual dance competition is toughly documented, where each recording is labelled with the performer name and choreography, which is categorized as 'Classical" or "Contemporary". We'll be specifically using footage from the *2014 Selections*.

'2014 Selections' dataset playlist: https://www.youtube.com/playlist?list=PL9Ep9acUXuI5et9Rh3XWggZee4g33MTUG
'Split Sides' choreography A and B examples: https://www.youtube.com/playlist?list=PLBGKFfYOol_M7OahQI0zQCNHoyUYt9eyC

1.6 Testing and Evaluation Criteria

To validate the plausibility of the results, we've constructed an online survey that asks users to rank and rate a set of outputs. For each stage, they are presented an example input segment, followed by the nearest neighbour outputs as well as a randomly selected segment from the video corpus. The user is asked to select a best and worst match, giving a 1 to 5 rating according to how well they think they capture the kinematic characteristics from the original video clip.

Table 1 Project Assessment

	Criteria	Assessment
1	We intend to reliably recognize movement qualities from video footage regardless of the performer or environment, extracting features that solely represent motion.	We obtain a diverse dataset of different performers of varying gender, age and ethnicity. We'll also examine the use data from different social contexts i.e. professional performance footage or amateur practice recordings. *See 1.5 for further details* During the feedback task, users will be comparing video segments from different performers
2	Where possible, we'd like to be able to identify demonstrations of individual LMA Efforts Elements when explicitly demonstrated.	The recordings from the LMA examples are segmented and embedding into the vector space where we study any meaningful clusters produced.
3	The system should be able to match segments where the same choreography is being performed by the subject.	The initial stages of the user feedback task will include examples of this type of match. We'll examine how consistently these are ranked as the "best match" along with the numerical rating provided
4	Where near-identical segments aren't available in the dataset, we should observe suitable pairings that capture some of the expressive characteristics of the input.	The latter stages of the user feedback will ask users to rank and rate a selection of pairings. If successful, we should see superior results for the computationally selected segments over those picked at random

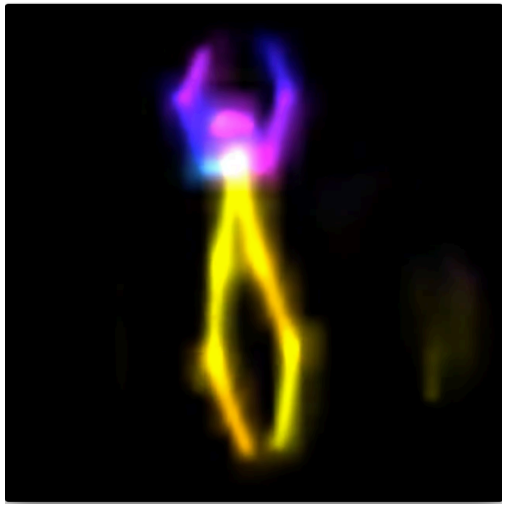

Figure 4 Part Affinity Field Render Output

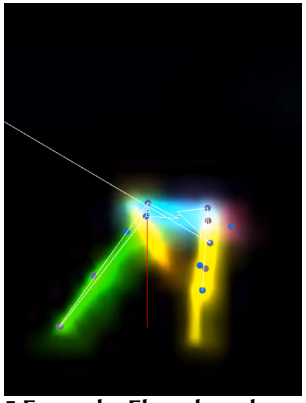

Figure 5 Example: Floor-based movement

2 Design Process

2.1 Extracting and visualizing OpenPose output

The original paper credits their accuracy to the use of Part Affinity Fields (PAF's). This provides a rich, non-parametric representation of the body part positions. For each frame, we get a visual representation of where human body parts are detected, colour co-ordinated in correspondence to their confidence rating. To retrieve the PAF visualisation along with keypoint data from the input footage, the following settings are applied to get the highest accuracy results as recommended by the developers:

```
$ ./build/examples/openpose/openpose.bin --video $media_dir --keypoint_scale 3 --
net_resolution "1312x736" --scale_number 4 --scale_gap 0.25 --disable_blending -
write_video output/$2/$2.avi --write_json output/$2/json --part_to_show 21 --display 0
```

Code Snippet 1 OpenPose Command

Where **$1** is the video path and **$2** is the output name, this calculates the skeletal keypoint and renders a video showing the PAF data. This would run on a on a single dedicated NVIDIA P6000 GPU, with 3840 CUDA cores.

The output is then visualised in a Processing sketch where the keypoint data is synchronized with the input video; this allows us to check the results for any major positioning and timing errors. It also helps us visualize and compare the effects of different features in real-time. By taking a test batch from our dataset and monitoring the output, we observe that the skeletal tracking was for the most part highly accurate where all the joints are valuable towards the camera. The system would usually struggle during more complex sequences, particularly those involving full body orientations (e.g. during a pirouette) as well as floor-based movements (see **Figure 5**). This was reflected by the significant drop in confidence values in the output, often leading positional glitches.

Regardless of the input, some positional jitters were consistently enviable, which would cause errors in global feature calculations (2.2.2). However, they were generally infrequent and could be filtered out with averaging.

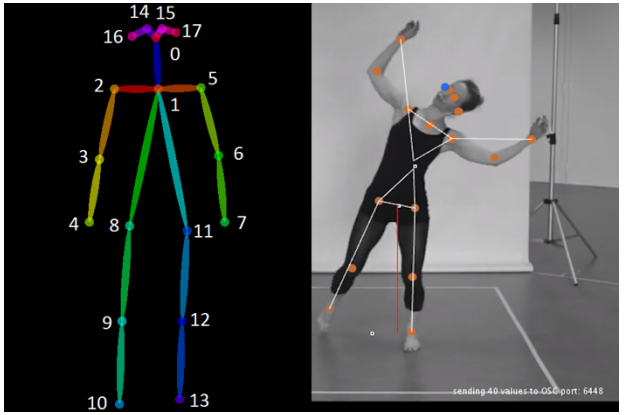

Figure 6 OpenPose output

(left) mapped to skeleton and synchronized with source video (right)

2.2 Biometric Feature Extraction

2.2.1 *Müller Features*

To represent the general motion of different parts of the body, we've adopted a subset of features from Meinard Müller's kinematic classification [21]. Müller describes a set of relational features that express intuitive and semantic qualities of a human pose. It is a well-balanced and efficient feature set designed to classify aspects of full-body motions. We've selected 20 of these features that could be suitably adopted to the 2-D skeleton data. These features are visualised along a continuous timeline as displayed in **Figure 7**.

In his literature, Müller proposes a set of binary and continuous features that can be used for gesture recognition. We focus solely on the continuous features with no intention to hard-code any definitions of each pose. Instead, we are just looking to the compare the outputs captured during each segment. **Figure 8** illustrates the key components of the following features:

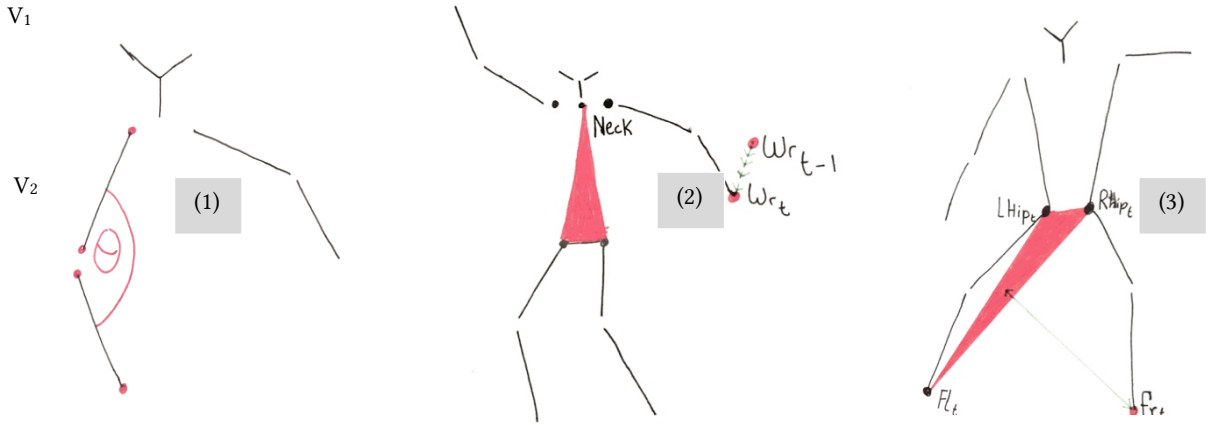

Figure 8 Müller Feature Sketches

(1) We examine the angle, θ between two vectors between the shoulder and wrist, informing the twist around the elbow.
(2) This is described as the velocity of the right wrist in relation to the neck in the perpendicular direction to the geometric plane illustrated in red. Where *t* represents the current frame, we find the distance between wrist positions during the current and previous frame (*t-1*).
(3) We measure the distance between the right foot and the entire plane occupied between the hips and the opposite foot; this is calculated by the point on the plane with the smallest distance from the foot.

For a complete list of the features used, see the link in the margin of this page **Error! Bookmark not defined.**. For further details, please refer to the full source code in section 8

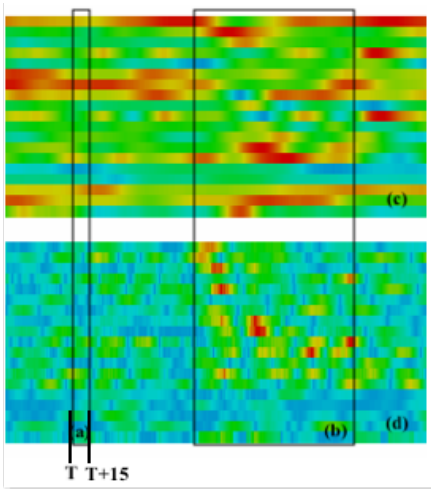

Figure 7 Data Visualization Example

The visualization consists of two parallel matrices (*c, d*). On the bottom, hue is mapped to the intensity of each Müller feature at a given frame interval (*T*), from blue (minimum) to red (maximum). From this segment, it's apparent where the subject transitions into a high intensity of motion (*b*), which in turn, contributes to the global kinematic energy

More annotated feature visualisations (along with corresponding footage) have been uploaded here: http://igor.gold.ac.uk/~wprim001/MOCO/final/

The full list of features used, along with their description can be found here: http://igor.gold.ac.uk/~wprim001/MOCO/final/featureClass/Muller.pde

2.2.2 EMA, Magnitude and Jerk – Kinematic Descriptors

The feature outputs, along with the raw skeletal positions are continuously put into circular buffers allowing us to compare the values between a set window of frames. Starting with the Estimated Moving Average (EMA) which smooths out the overall output; this helped filter some of the positional glitches apparent in the raw output. From the smoothed output, we calculate the magnitudes (rates of change) which indicates the power applied to each bodily function. The sum of each feature magnitude is accumulated into what we'll refer to as the *Total Movement Magnitude*, which is calculated at each frame.

$$\boldsymbol{TMM}(t_i) = \sum_{k=0}^{n} \frac{\Delta F^k(t_i)}{\Delta T} \tag{1}$$

We can also calculate the 'jerkiness' of a particular feature or joint. This indicates the smoothness of motion from the change in acceleration [24]. In the MotionMachine workshop [20], Tilmanne et al. describe how they used the weighted average jerk to estimate the 'directness' of the overall motion, associated with the LMA Effort, Flow. We can apply this to the feature outputs like so:

$$\boldsymbol{jerk}(F^{\boldsymbol{k}})(t_i) \frac{F^k(t_i) - 2.F^k\left(t_{i-\frac{w}{4}}\right) + 2.F^k\left(t_{i-\frac{3w}{4}}\right) - F^k(t_{i-w})}{2.\Delta t^3} \tag{2}$$

Where F^k is the feature output at frame t_i, and the window size is set as w. For more details, please refer to the **dataBuffer** class in the source code (Section 8).

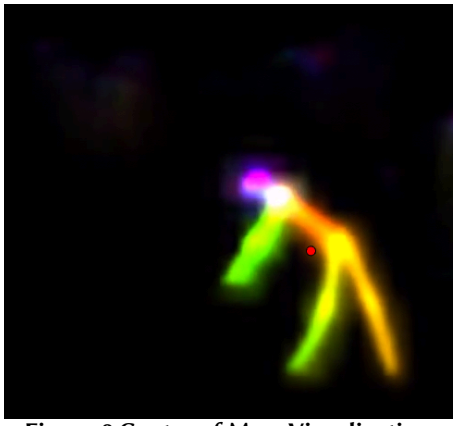

Figure 9 Centre of Mass Visualization

The bodily centre of mass doesn't necessarily have to be attached to the individual. For example, in **Figure 9** the centre of mass is positions in front of the torso, indicated by the red circle

2.2.3 *Weighted motion features*

Previous studies have provided standard anthropometric data, including a set of relative measurements and mass data for individual human body parts **Error! Reference source not found.**. These values can be applied to the skeletal joint positions to estimate the bodily centre of mass. This measure can be used to depict high-level descriptors regarding balance and weight distribution (postural load).

$$\boldsymbol{X(COM_i)} = X_{prominal} + (Length\,\%)\,(X_{distal} - X_{prominal}) \quad (3)$$

Where *i* refers to the body part between the proximal and distal joint, we retrieve the location of the segment's centre of mass [27]. Each segment is assigned a mass percentage in proportion to the total body mass. Each mass percentage is multiplied to its position and accumulated towards the bodily centre of mass.

$$\boldsymbol{CoM_{global}} = \sum_{i=1}^{N} w_i \,.X(COM_i) \quad (4)$$

Where w_i is the weighted mass of the considered body segment *i*. The COM position would represent the total distribution of the individual's mass during motion. However, this was dependant on all the key point outputs, any outliers or undetected joints would result in misleading COM calculations. From here, we decided to use the same segment weights to express the *Quantity of Motion*.

$$\boldsymbol{QoM}(t_i) = \frac{\sum_{k=1}^{K} w_k . v^k(t_i)}{\sum_{k=1}^{K} w_k} \quad (5)$$

The *Quantity of Motion* is defined by the weighted average speed of all *K* joints, the single value indicates the performers change in weight distribution within a given window of frames.

Classification of LMA Efforts – Segmentation Approach #1

Using the LMA example dataset, we implemented four (4) K-nearest neighbours (KNN) classifiers that are used to categorize the four LMA movement effort qualities: Flow, Weight, Time and Space independently, where each Effort Factor operates on a continuum between two classes. In our tests, we run each of these classifiers separately and compare the results. Note that in this experiment, we only used the biometric features.

Table 2 LMA Effort Identification Results

Effort Quality	Number of frames trained	Calculated cross-validation accuracy (10 folds)	Test data success rate (% of correctly classified frames)
Free	2528	99.72%	84.27%
Bound	2094	99.72%	73.82%
Sustained	2505	99.84%	75.53%
Quick	1807	99.84%	69.83%
Light	3435	99.93%	72.93%
Strong	2262	99.93%	63.01%
Indirect	2435	99.92%	50.29%
Direct	1460	99.92%	57.37%

In the interim report[26], we discuss the varying success rates when trying to identify each LMA Effort from video. We experienced a few issues due to the lack of fidelity and quantity of training data, which only contained footage of one performer. When evaluating the output, we noticed that the incorrect labels were usually applied over segments significantly shorter than the rest, usually during transition inconclusive of any definite description. Even when the labelled output would successfully coincide with that of the ground truth, the segments were liable to contain incomplete gestures.

Another issue with the dataset was that the footage only demonstrates singular components of Effort, where observable gestures are characterized when these elements are grouped and arranged into states, drives and actions, relating to the emotional traits conveyed in a physical gesture.

Whilst further developments could have been made to improve the accuracy of the continuous classification, this didn't present itself as an effective method for automating segmentation, particularly for new, unseen footage. It was also apparent that we'd need to explore the use of other features to gain more insight into the qualities of movement recorded.

2.3 Period of Kinematic Exertion - Segmentation Approach #2

In this approach, we accumulate frames that represent a single observable gesture, splitting the source footage into 0.5 to 5.0 second *Periods of Kinematic Exertion (PKEs)* which can be defined as a continuous sequence of physical effort. This determined by the *Global Kinematic Energy*, calculated by the *Quantity of Motion* and the *Total Movement Magnitude* (see 0)

$$\boldsymbol{KE}_{global}(t_i) = \boldsymbol{QoM}(t_i) \bullet \boldsymbol{TMM}(t_i) \quad (6)$$

Where time is given at frame T of the video and F^k represents each Müller feature output. When the global kinematic energy crosses a constant threshold value, a new segment is rendered between the end of the previous segment and the current frame, with a maximum duration of 5.0 seconds. The exported feature recordings can then be split according to the segment frames.

With this hard-coded method, we experienced a much greater consistency in the segmentation throughout our entire dataset. Whilst this didn't provide any means of labelling the output, the segments clearly displayed a complete gesture, each exposing distinct kinematic properties. When applied to the repeated choreographies in the *Split Sides* foot
age, we notice that the segments would often show the same movements with near-identical start and end positions, making them suitable benchmarks during our tests.

Periods of Kinematic Exertion
A *Period of Kinematic Exertion (PKEs)* can be defined as a continuous sequence of physical effort

```
for file in imageBuffer:
    img = file
    if img is not None:
         #layer the following frame
        combined_heatmaps =
        cv2.addWeighted(prevImg, 1,
        img, 1/10, 0)
        #copy the result and repeat
        prevImg = combined_heatmaps
```

Code Snippet 2

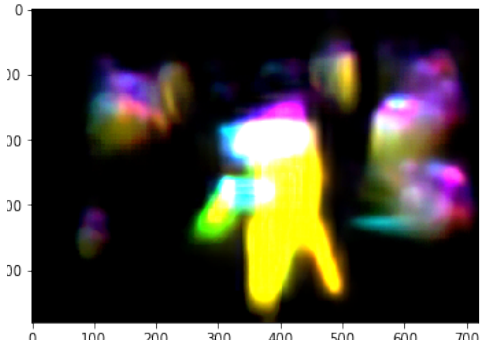

Figure 10 Layered PAFs

This static representation informs the dynamic arrangement of the performer's limbs within their own kinesphere [25]. Just from eyeballing the outputs, we can recognize some aspects of their physical composition. However, we are required to filter out the patches of noise before applying this to a computational model.

2.4 Visual Feature Extraction

The biometric features described in section 0 represent the dynamic/kinematic characteristics of the recorded movement. However, they don't reveal aspects of the subject's spacial qualities, whether that be relative to themselves or towards their surrounding environment. Whilst these qualities can be defined using bounding shapes and convex hull analysis [24], which depend on skeletal information, we focus on the visual features of the Part Affinity Fields(PAFs) detected by OpenPose. For each frame, the PAF output isolates the bodily figure from the rest of the image. A collection of outputs are assigned to each video segment.

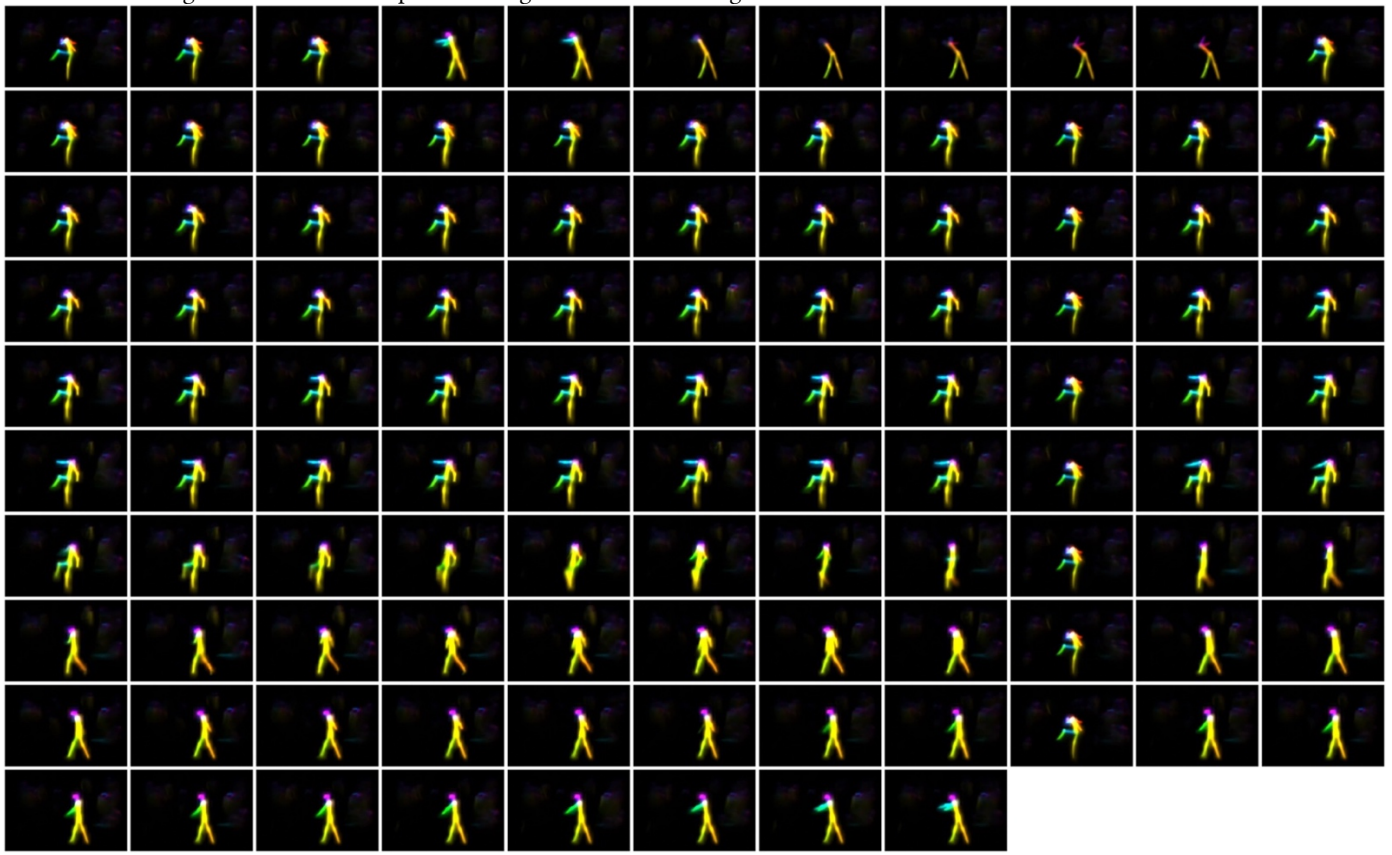

Figure 11 Individual PAFs of segment (107 frames)

Figure 11 displays a 3.57 second sequence of motion that comprise a PKE. Each frame is layered at 10% of its original brightness (see **Code Snippet 2).** This resolves the output shown in **Figure 10.**

It's important to keep the physical shape intact when we scale the final image. To avoid overstretching the image on the x-axis, we apply a small amount of padding relative to the width of the bounding box.

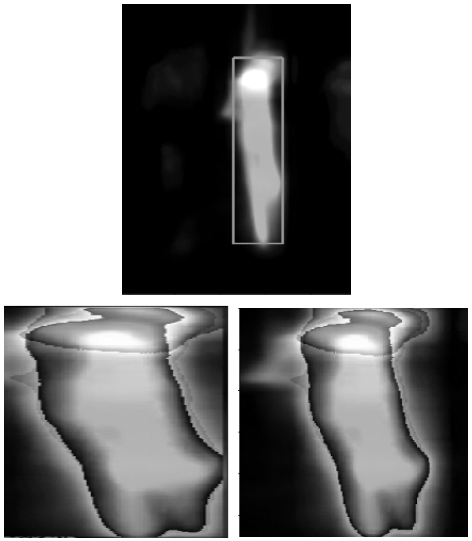

Figure 13 Narrow Pose Cropping

Top: Narrow bounding box

Bottom-left: Overstretched Output

Bottom-right: Output with padding

Contour recognition is applied to the output to define a bounding box around the human figure. **Figure 14** demonstrates the result of applying a threshold, clearing pixels below a given intensity (set at 155) in the greyscale image, followed by a morph kernel to filter out any remaining noise.

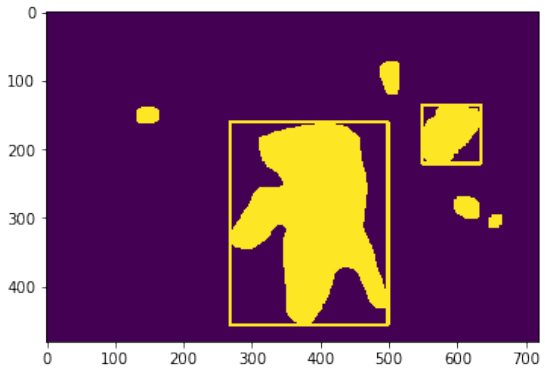

Figure 12 Bounding Boxes

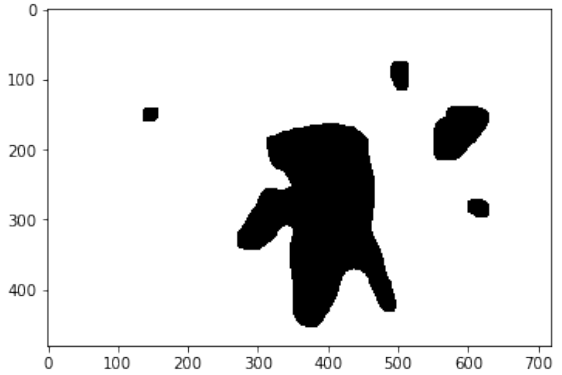

Figure 14 Filtered Heatmap

Form here were outline all of the contours in the image and compare the areas of each bounding box. Provided they are in a given range, set in relation to the resolution of the input, we crop the layered heatmap accordingly. Finally, we add the output of the first frame to outline the opening pose, before scaling the image down.

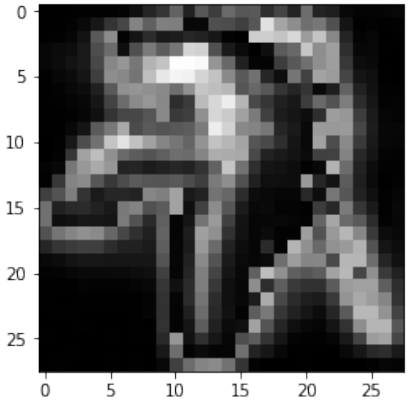

Figure 15 Cropped Heatmap

The result consists of 28x28 pixels each holding a single brightness value. This format is comparable to that of the MNIST dataset [11] which is a commonly referenced benchmark for testing machine learning models as its format makes it highly suitable for Computer Vision and pattern recognition

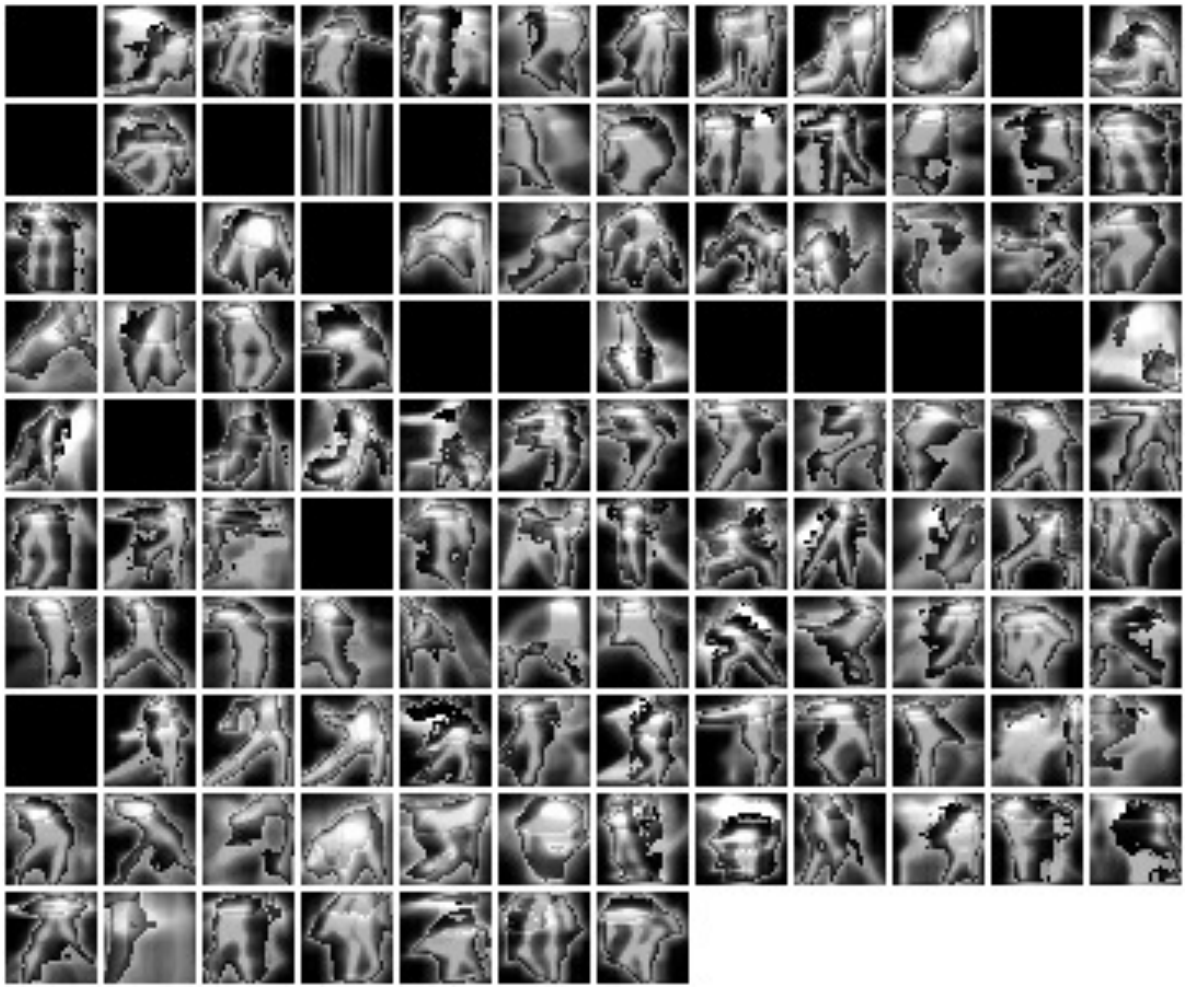

Figure 16 Condensed PAF Outputs

Above, we have collected 115 outputs from the segments of a single choreography. The fully black results demonstrate where no subject is present, or the automated cropping was unsuccessful. This would often happen during camera angle transitions, leading us to collect footage with (mostly) constant camera placements throughout.

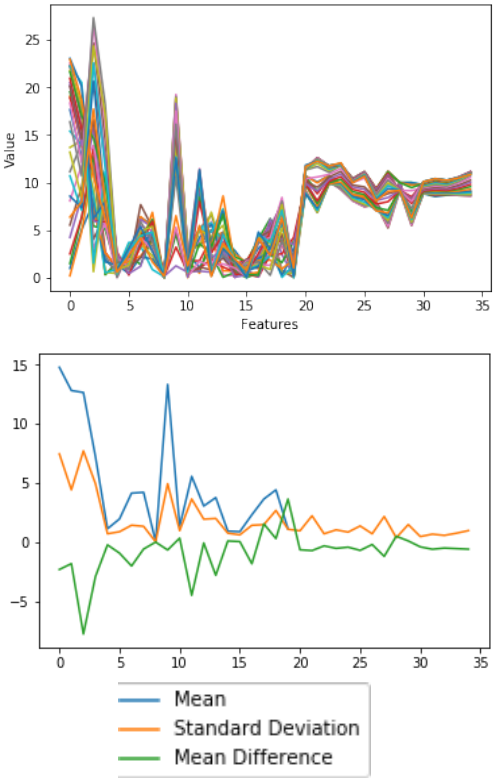

Figure 17 Feature Vector Consolidation

On the top, all 35 features are plotted for each the frame in the segment. This is then reduced down to just 3 vectors. A sample of feature matrices can be found in **Figure 21**, giving a clearer representation.

3 IMPLIMENTATION

3.1 Consolidating Biometric Features for Individual Video Segments

The scale of the exported biometric feature matrixes is dependant of the length of the video segment, determined by *n* frames. The matrix is split between the continuous Müller features, which are geometrically normalized and the weighted skeletal positions. In this example, we've recorded the magnitudes of the 20 Müller feature outputs along with a soothed average of 15 joint positions. All 35 features contain *n* inputs, which are encoded into single values, normalized according to the respective duration of a segment using the following numerical functions:

- Mean: Calculates the average reading throughout the sequence. Note that this is only taken for the Müller features.
- Standard Deviation: Computes the spread of the biometric outputs. A high deviation can be associated with large-scale gestures.
- Mean first order difference: For each feature, we calculated the mean difference between the successive frames of the segment. This correlates with how much features changed from start to finish.

This approach was influenced by a project that applies these functions to concatenate the perceptual features of drum samples, irrespective of length [29]. The same method can be adapted to include jerkiness and other arrangements of inputs. However, we found the above example provided a sufficient representation of the kinematic energy distributed around the subject's body, with just 90 values.

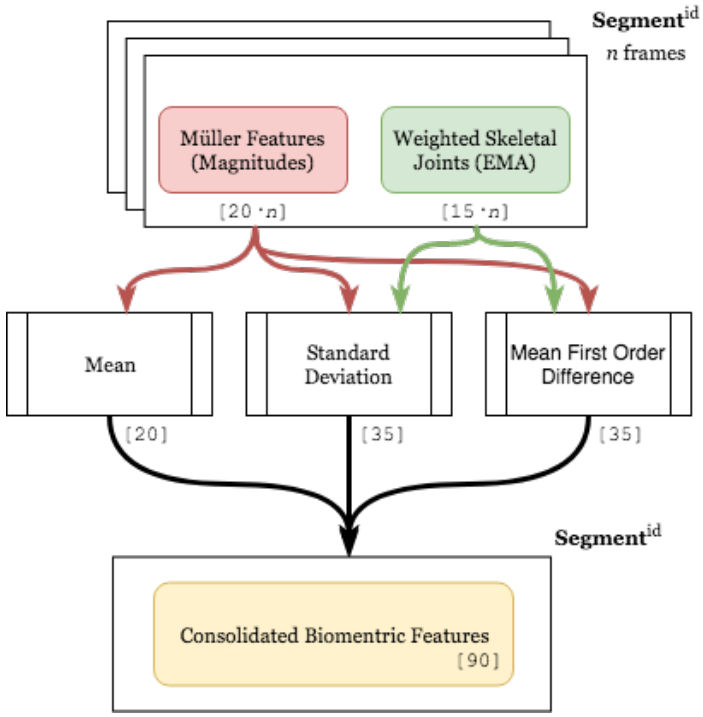

Figure 18 Feature Consolidation

3.2 Static Visual Feature Representation - Histograms of Oriented Gradients (HOGs)

The pre-processed heatmaps each contain 784 pixels. Using the raw data as an input would be inefficient as we'd be supplying a lot of redundant data. Instead, we take the outlining show of the figure using a Histograms of Orientated Gradients (HOG).

> *"The histogram of oriented gradients (HOG) is a feature descriptor used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image. This method is similar to that of edge orientation histograms, scale-invariant feature transform descriptors, and shape contexts, but differs in that it is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved accuracy"* [33]

The HOG descriptors are particularly effective at modelling the structural and textural qualities of an image for the way it captures the magnitude and direction of the edges. They are commonly used towards object recognition tasks and prove to be work well with machine learning classifiers.

```
def get_hog() :
    winSize = (28,28)
    blockSize = (8,8)
    blockStride = (4,4)
    cellSize = (8,8)
    nbins = 9
    derivAperture = 1
    winSigma = -1.
    histogramNormType = 0
    L2HysThreshold = 0.2
    gammaCorrection = 1
    nlevels = 64
    signedGradient = False

    hog = cv2.HOGDescriptor(
      winSize,blockSize,blockStrid
      e,cellSize,nbins,derivApertu
      re,winSigma,histogramNormTyp
      e,L2HysThreshold,gammaCorrec
      tion,nlevels,
      signedGradient)
    return hog
```

Code Snippet 3 Histogram of Oriented Gradients Configuration

These parameters were influenced by the recommendations suited towards pedestrian detection[32] as well as MNIST embedding [33]. The unsigned gradients return angles within the range [0-180]. We also allow 9 orientation possibilities which considers increments of 20 degrees.

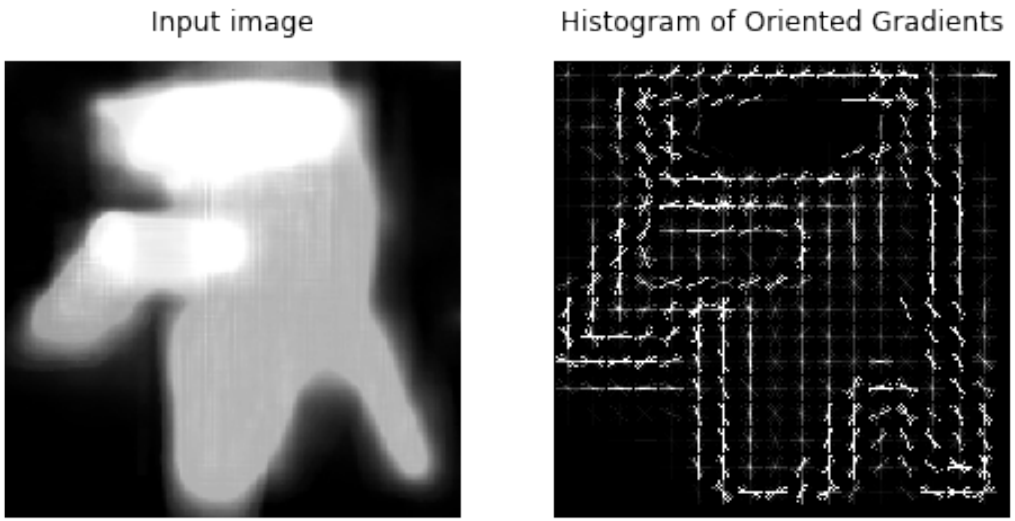

Figure 19 Histogram of Oriented Gradients Example (Higher resolution)

With the parameters listed in Code Snippet 3, our input image is split into cells of 8x8 increments, which each return an intensity and angle collected from the windowing kernel. This accumulates to a feature vector of 324 values, meaning we've managed to reduce our dimensions by half whilst providing a more suitable reorientation of the subject's spacial attention.

3.3 Embedding Data – PCA, t-SNE and UMAP

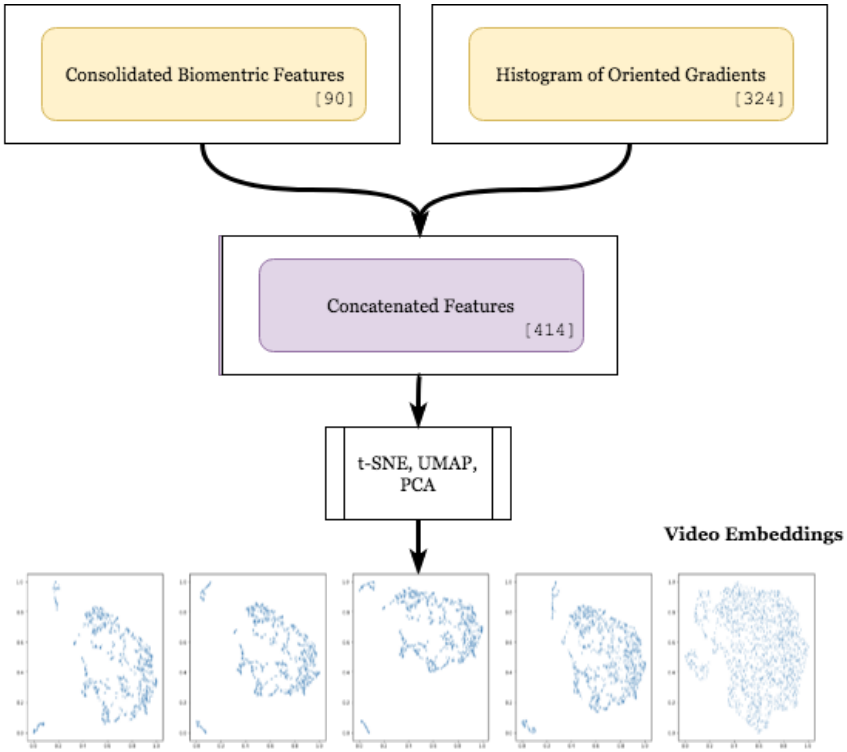

Figure 20 Preparing Dataset
The consolidated feature sets are consolidated and processed through t-SNE, UMAP and PCA dimentionality reduction algorthims to produce video embeddings that can be explored in 2D space.

The use of t-SNE and PCA are well established and there is a lot of information regarding existing projects that incorporate these methods for dimensionality-reduction [30]. On the other hand, Uniform Manifold Approximation and Projection (UMAP) has recently been introduced into the machine learning community [31]. Compared to t-SNE, the developers praise the improved speed for which UMAP is able to process its outputs, making it highly suitable for large datasets. Furthermore, it claims to provide better reservation of global structure within the embeddings. We test the t-SNE and UMAP algorithms with a range of parameters regarding how it structures these plots. The results of these are discussed in section 4.2.

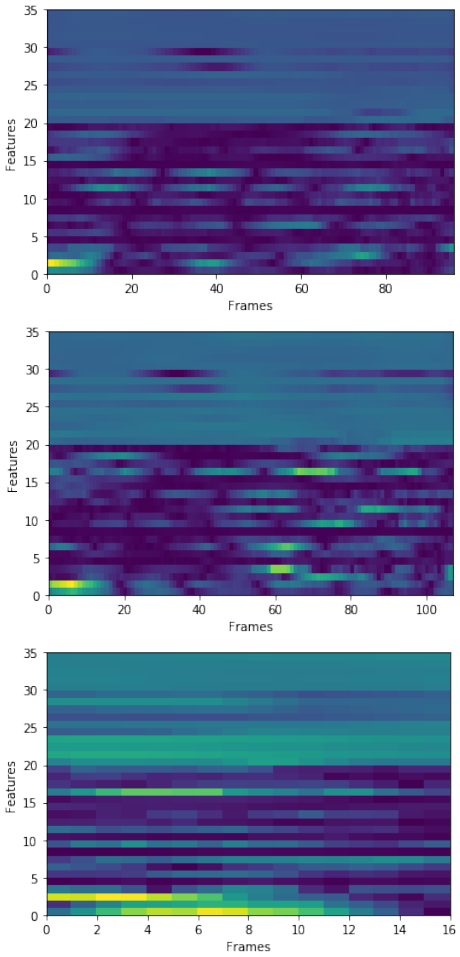

Figure 21 Biometric Feature Comparisons

The top two matrices were taken from the same choreography and the bottom was selected at random

4 APLICATION

4.1 Input Feature Comparison

By examining the features from similar segments, and comparing against those from a random sample, we can make suitable judgments towards the expected output. Two matching clips from the *Split Sides* solos, displaying the same movement are used as a benchmark. The Structural Similarity Index is a method that models the perceived differences within the structure of an image. This can be applied to our layered PAF outputs.

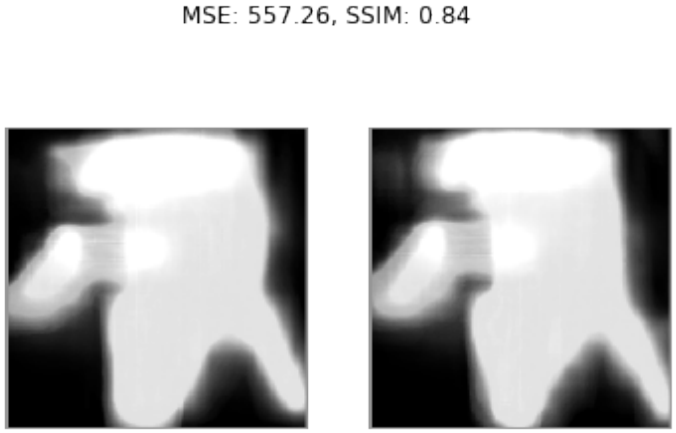

Figure 22 SSIM Comparison #1

In **Figure 22**, the overall shapes appear very similar. Where an SSIM index of 1.0 indicates perfect similarity, the structural similarity is relatively high at 0.84. When compared to the output from the random sample, the SSIM index drops to 0.43.

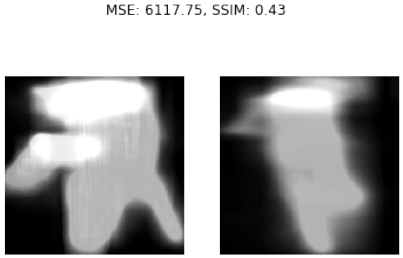

Figure 23 SSIM Comparison #2

'Split Sides' test examples: https://www.youtube.com/playlist?list=PLBGKFfYOol_M7OahQI0zQCNHoyUYt9eyC

Reviewing Consolidated Features

It's interesting to note that, even when the movements appear identical in the source footage, the precision of these initial feature outputs expose very subtle differences in each clip. If we compare the consolidated inputs for these segments however, we notice a stronger relationship in the data.

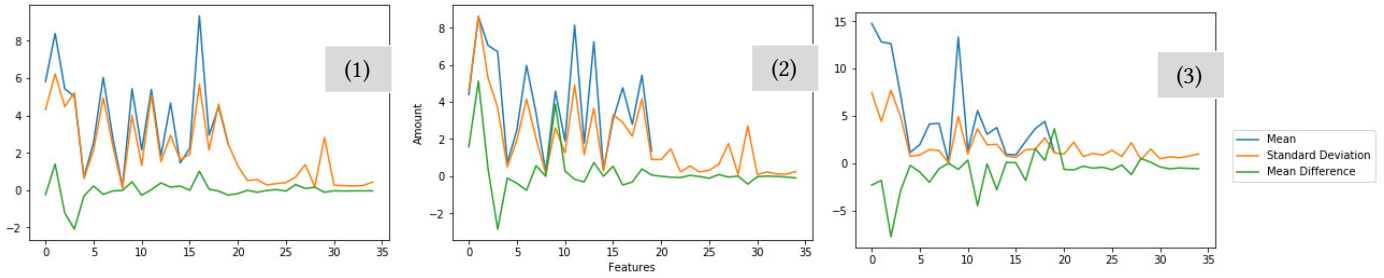

Figure 24 Consolidated Feature Comparison

Plots **1** and **2** display the three time-independent biometric feature vectors from *Split Sides*, and **3** is taken from a different choreography. At a glance, it's possible to recognize the differences in overall structure between these plots, suggesting a radical difference in the kinematic content in the random sample.

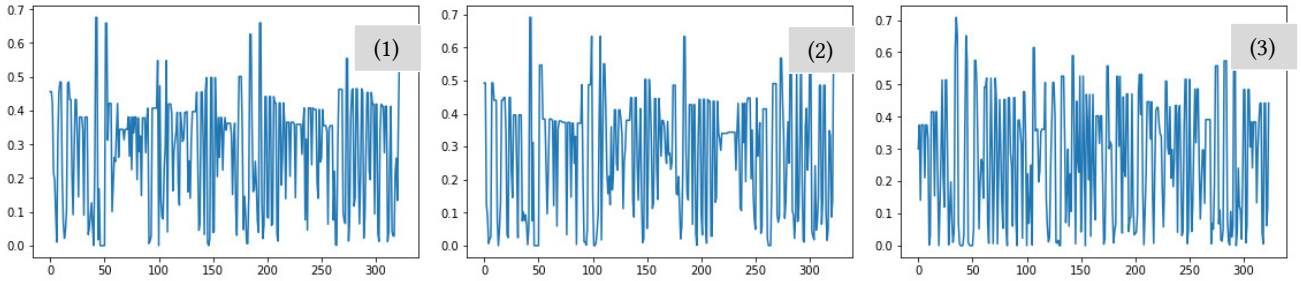

Figure 25 Histogram of Oriented Gradient Plots

Figure 25 compares the HOG vectors for the same segments, informing the subjects geometric attention. The crossover is harder to observe here, so we calculate the distances between the vectors, resolving to 2.24 and 4.16 between **1** and **2** against **1** and **3** respectively.

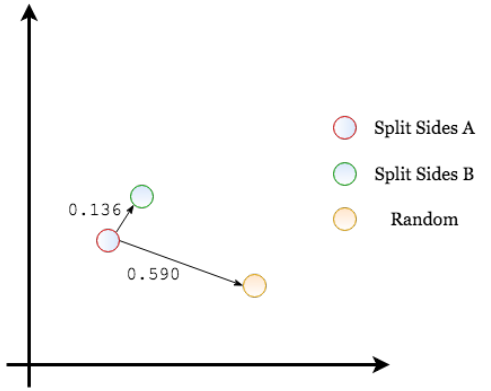

Figure 26 UMAP Distances

The distances for the test inputs are recorded from a UMAP embedding, with the neighbour and distance metrics set to 5 and 0.1 respectively, conjuring a focus towards local clusters. In this test, we find the segments from the same choreography have a smaller Euclidian distance of 0.136 compared to the random sample.

When changing the UMAP parameters, these distances were mostly similar. The main differences could be observed in the global structure, which we discuss in section 4.2.

4.2 Feature Embedding

From here, we can assume that most of the mimicking segment pairings would be plotted near one another. Although, when observing the final plots, the consistency of these matches was highly dependent on the parameters of the dimensionality-reduction algorithms, impacting the sparsity of the local structures. Mimicking segments were more likely to cluster where the neighbour or perplexity indexes were lower, though the transitions between clusters could be perceived as more dramatic, suggesting an abstraction of the global representation of the dataset. The influence of the structural parameters are shown in the plots below, starting with the UMAP embeddings. Each row comprises results of selected minimum distance values incremented from 0.0 to 0.5 and the columns represent the number of neighbours within the plot.

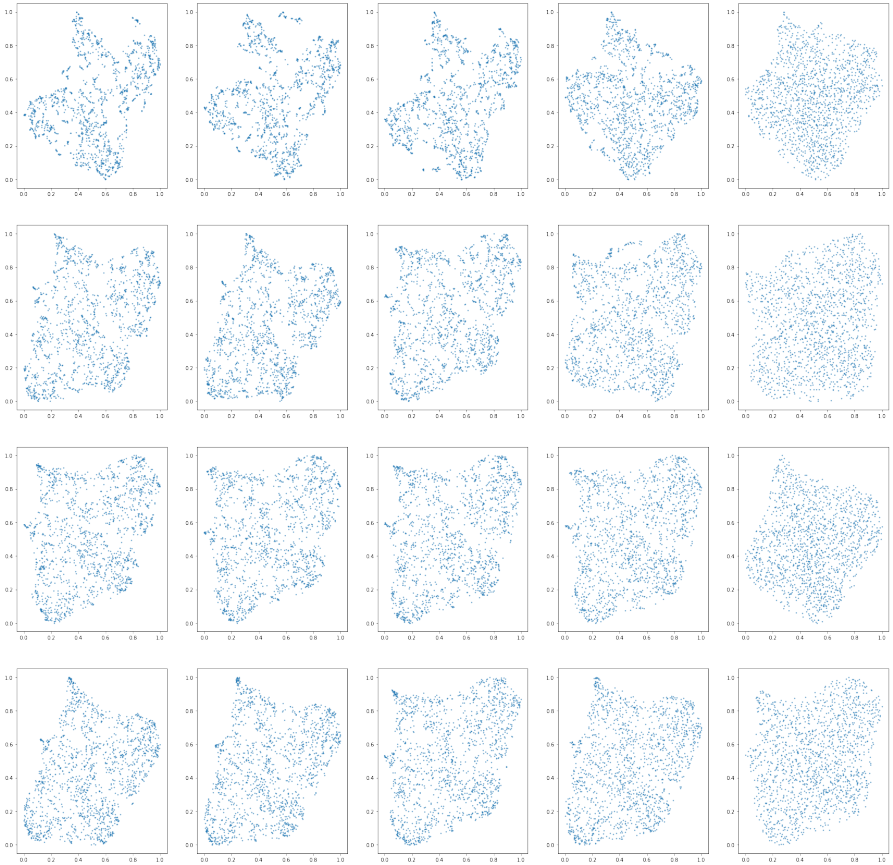

Figure 27 UMAP Embeddings, Biometric Features Only

We tend to set our neighbouring amounts and perplexities relatively low, as our corpus was limited. At best, we wouldn't expect more than 5 plausible segments pairing in a single cluster.

As we increased the number of neighbours the defined local clusters would eventually disperse into a single large-scale arrangement. These plots weren't so useful as it was difficult to perceive any correlation within the global structure compared to the content of the immediate neighbours.

Where some clusters were densely populated, we could adjust the minimum distance accordingly to help us explore the individual segments when using the interactive visualizer

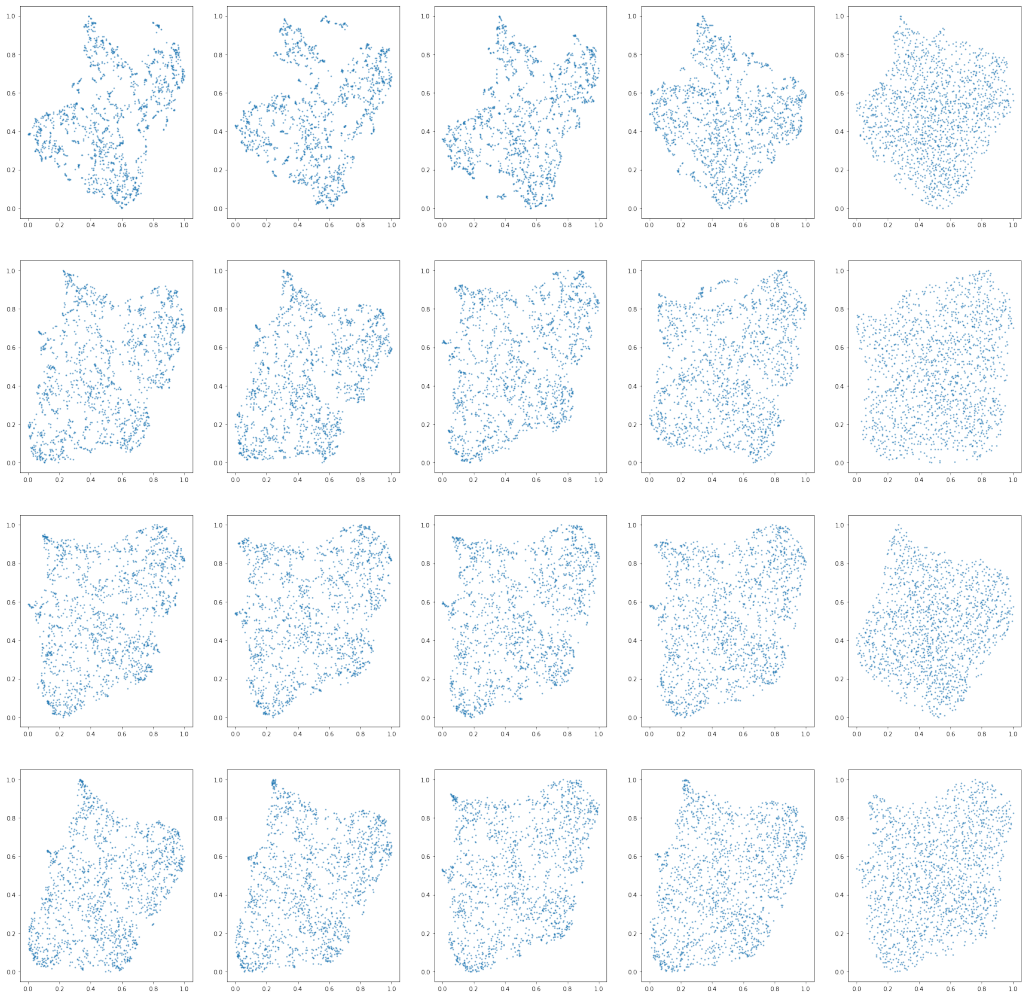

Figure 28 UMAP Embeddings, Visual and Biometric Feature Sets

The parameters in these examples are as follows:
Minimum Distance (row) : 0.00, 0.001, 0.01, 0.1, 0.5
Number of Neighbours (column) : 5, 10, 15, 30, 50

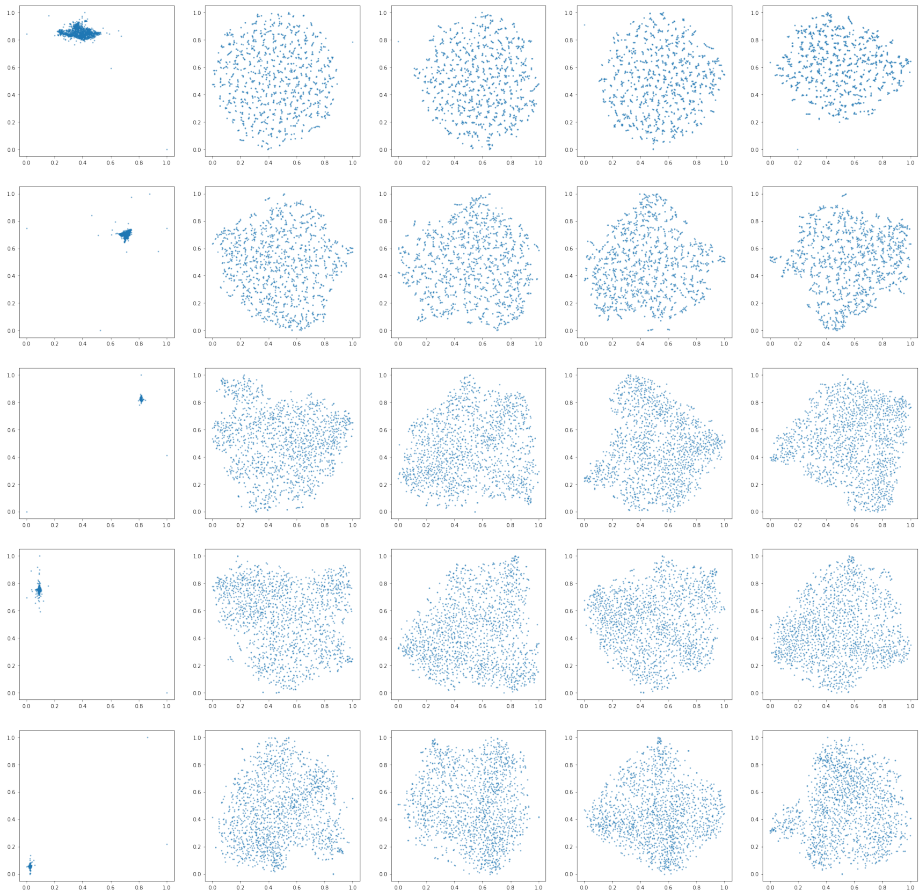

Figure 29 t-SNE Embeddings, Biometric Features Only

It was apparent that only by only the biometric features to the model, our ability to recognize clear matches within the data was limited. When the visual features are included, we can begin to distinguish local structures given enough iterations. We found that 1500 iteration would provide consistency within each cluster

Due to the extensive processing time required by the t-SNE function, we cap our iterations at 2000 for larger datasets.

An insufficient perplexity would squash all the data into an indecipherable space. Once a clearer global structure was formed, increasing the perplexity further would often just rearrange the same clusters into different global positions.

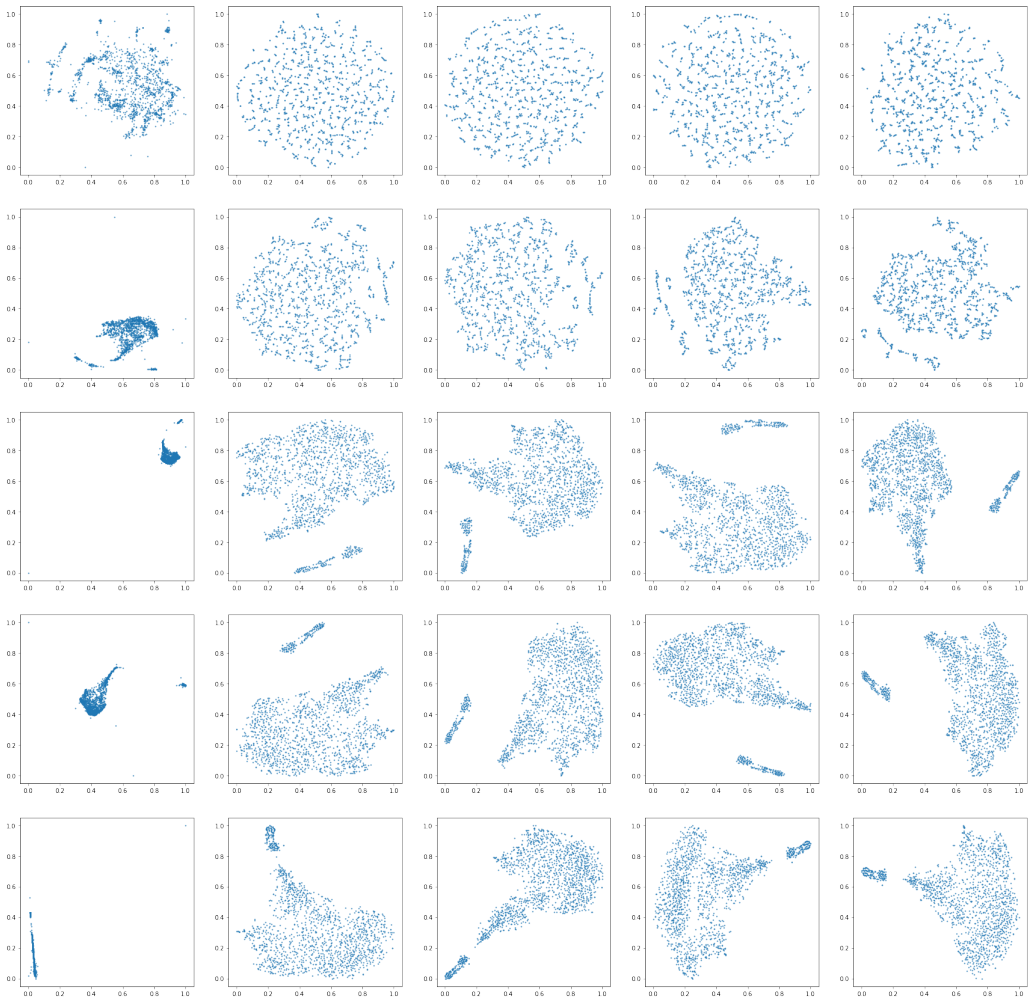

Figure 30 t-SNE Embeddings, Biometric and Visual Feature Sets

The parameters in these examples are as follows:
Number of Iterations (row) : 250, 500, 750, 1000, 2000
Perplexity (column) : 3, 5, 30, 50, 75

4.3 Interactive Visualizer

The plots were imported into an interactive visualizer, allowing the user to assess the results from different algorithms and parameters. Upon navigating the plots, we're able to compare the footage contained in the segment, either as raw video or from the PAF output. We separated our results to those generated from the biometric features only and those that apply the full feature set. You can find links to both of these in section 8.

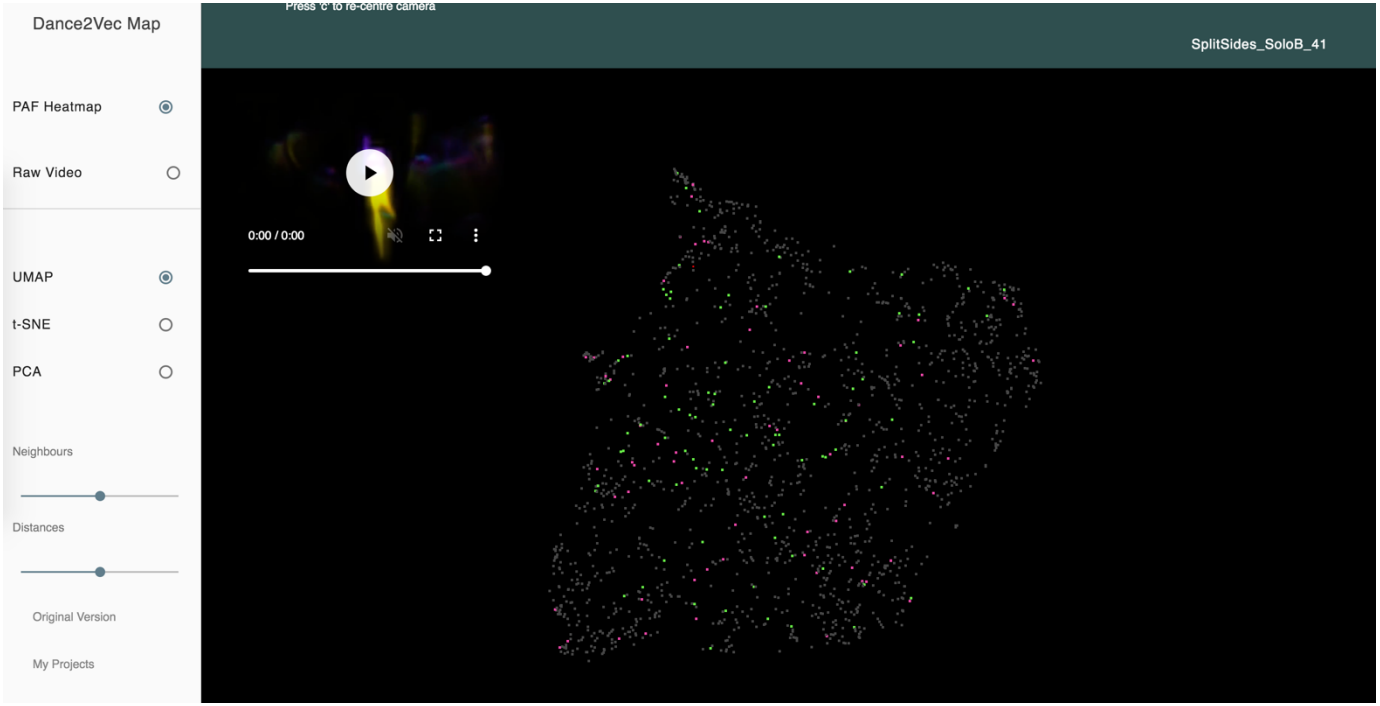

Figure 31 Interactive Visualizer

5 RESULTS AND DISCUSSION

5.1 User Feedback Test

In the following section, we'll evaluate the results from our user survey[6], asking participants to judge the perceptual similarities between some examples of neighbouring segments selected from the t-SNE plots. The task is split into 8 stages, taking 4 examples that only account for the biometric features and 4 examples where the visual/geometric features have also been applied. We received a total of 23 responses.

Table 3 Individual results for each stage

Stage	*Feature Set*	*% ranked neighbouring segment as best*	*Mean rating of 'best match' (1-5)*	*% ranked random segment as worst*	*% agreed on the same best match*
1	Biometric + Visual	100.0%	4.43	69.6%	100.0%
2	Biometric	73.9%	2.43	43.5%	26.1%
3	Biometric + Visual	100.0%	4.52	69.9%	100.0%
4	Biometric	100.0%	3.83	69.6%	95.7%
5	Biometric + Visual	100.0%	2.61	65.2%	65.2%
6	Biometric	100.0%	2.91	95.7%	60.9%
7	Biometric + Visual	95.7%	3.35	73.9%	82.6%
8	Biometric	91.3%	2.65	34.8%	56.5%

At each stage, the user is presented with an input segment along with 4 clips that are supposed to imitate it. 3 out of four matches are selected from the nearest neighbours to the input, the other is taken at random; the matching segments are then presented in a mixed order. Users are asked to pick out a 'best match', provide a similarity rating and then choose a 'worst match'. The segments are displayed as their PAF heatmap representations to focus the participants attention towards the appearance of the motion.

The percentage of the agreed best match indicates where one particular pairing would stand out better than the rest. In the case of the **1st** and **3rd** stage, all participants agree on the same option where a near-identical segment is retrieved. During the other stages, we can assume

The example results used in this test were obtained using a premature method of formatting the input data. Before applying the Histogram's of Orientated Gradients, all 784-pixel values we fed into the dataset. This caused a noticeable scale of inconsistency and error that has since been improved.

Dance2Vec test survey: https://patch-cupcake.glitch.me/

that even though the rankings were spread out, participants almost always opted for the neighbouring segment. However, the mean similarity ratings during these stages were far lower. The segments provided in stage **2** were overall poor matches and it was not clear what was supposed to be similar. This reflects the faults that frequently occurred when we only applied biometric features.

5.2 Feature Set Comparison

Table 4 Average Results

Feature Set	*Average % ranked neighbouring segment as best*	*Average Mean rating of 'best match' (1-5)*	*Average % ranked random segment as worst*	*Average % agreed on the same best match*
Biometric + Visual	98.9%	3.72	69.7%	87.0%
Biometric	91.3%	2.96	48.9%	59.8%

In

that even though the rankings were spread out, participants almost always opted for the neighbouring segment. However, the mean similarity ratings during these stages were far lower. The segments provided in stage **2** were overall poor matches and it was not clear what was supposed to be similar. This reflects the faults that frequently occurred when we only applied biometric features.

5.3 Feature Set Comparison

Table 4 we display the average results retrieved from either feature set. When we compare the results, we observe that inclusion of the visual features gained better agreeability and similarity ratings towards their neighbouring segments, whether that be from the same or different choreographies. Where the biometric features usually managed to retrieve at least one segment of somewhat plausible similarity, the participants were less successful in finding a distinct odd-one-out. This coincides with our observations when exploring the local clusters, which often displayed inconsistent matches within the same area.

Where the combined features were somewhat effective in matching near-exact mimicking of the input, the visual feature set gained more plausible results while clustering segments from different choreographies. In the case that a near-perfect match was unavailable in the video corpus, the content of the neighbouring segments often portrayed similar characteristics regarding the subject's spacial orientation.

5.4 Considerations and Perspectives

The t-SNE UMAP and PCA algorithms can be parameterized to produce 3D embeddings. During development, we compared the Euclidian distances between the neighbouring clusters in the 2D and 3D t-SNE plots where we observed similar relationships towards a test batch of input data. Other projects that implement 3-dimetional embeddings report richer local structures [30] as the embeddings have an additional dimension to segregate from one another. In a lot of our 2D plots, we noticed a lack of clarity between the content of adjoining local structures. A 3D plot may improve the separation of these clusters.

The neighbouring segments used for these tests were based on an unbalanced representation of the visual features. The results here encouraged an exploration on applying structural features of the heatmap images. This lead to the utilization of HOGs, as described in 0, which significantly improved our model. Furthermore, the tests included a limited range of mostly high-quality results that could be easily interpreted by the users. However, it was apparent that amongst the global video segment embeddings, the plausibility of the clusters varied quite dramatically depending of the style of movement. For more credible results, we'd develop the test to include far more results. The dataset used in this test comprised 30 videos totalling in 2569 seconds of usable footage. After developing the system to achieve more consistent results, a reasonable next step would be to consider the impact for using more data.

The restrictions imposed by the way our features are represented independent of time were evident during implementation process. For instance, the biometric feature vectors in their full per-frame representations, are potentially well suited towards modelling one's kinematic attributes throughout the segment. However, the process in which we consolidate these into single values for each segment abstracts time-based information, such as periodicity or progression. Nevertheless, this method was effective in modelling the overall energy applied to different parts of the body.

Additionally, A successful match of segments was noticeably dependant on the similarity of the initial pose during the starting frame, largely relying on the video segmentation taking place at the same point in motion. A sliding window approach, independent of frame order should be more effective in matching movement qualities, regardless of exact timings. Although, this would require significantly more computation time, making it unsuitable towards large datasets.

5.5 Overall Reflection

On the whole, users were successfully able to tell neighbouring segments apart from a randomly selected clip, indicating the content within clusters were to an extent, perceptually resemblant. However, the ratings towards the most favours matches were often quite low (floating between "somewhat complimentary" to "little resemblance") compared to the higher scores given to parings from matching choreographics. This might suggest that the model wasn't able to find the most suitable match amongst the entire dataset, or that there wasn't enough data available in the corpus to provide anything better.

In its current state, this system demonstrates some potential finding accurate parings when given data from the same choreography, though with limited reliability, making it less effective for applications such as a reverse video search. On the other hand, we realized that we can utilize these tools to explore perceptually similar segments; this might be useful as a choreographic tool that might inspire new transitions between gestures. On the whole, this approach supports the exploring movement qualities from a wide range of datasets.

During the design process, we moved on from trying to associate our outputs with LMA Effort attributes and adopted an unsupervised-learning approach. As a result, our structures failed to provide much correlation regarding purely expressive descriptors of movement. A further study, supported by expertise within movement theory and psychology, would focus on adapting the model to interpret the emotional states projected onto the user.

Regarding the implementation into real-time interactive machine learning applications, there are still a few limitations regarding overall computation time. All our data was pre-processed within the OpenPose frameworks, taking approximately 20 minutes per minute of video footage on a single dedicated NVIDIA P6000 GPU. However, the framework is capable of operating in real-time, with lower accuracy, but still sufficient as an input modality. The UMAP and PCA algorithms could produce results from multiple parameter setting almost instantly when applying our largest dataset, composed of over 1500 elements of 434 total dimensions. The t-SNE embedding took considerably longer but still adequate for efficient development and tests. However, to process a new stream of data, the algorithms would need to run again from scratch.

For public review, the interactive data visualizer was posted onto an online forum titled "Movement Culture", it received some written feedback from one of the users:

> *"I believe that there are great possibilities of utilization of such a tool as a means to 'translate' movement-based compositional qualities into other types of composition such as musical or architectural. I, for one think it has a lot of potential."* ***Giannis Dimopoulos***

This strikes as an interesting perspective, relating the output to other formats. Where there already exists a range of projects that use dimensionality reduction to visualize features of audio[14], a future development could consider matching the clusters of movement qualities with relatable sounds (or vice versa), potentially enhancing the user's appreciation and understanding of the two.

6 CONCLUSION

This project introduces a pipeline that produces video embeddings based on observable movement qualities, using an unsupervised learning approach for dimensionality-reduction. We review a set of methods for extracting normalized features, allowing the system to find similarities in movement qualities, irrespective of the subject's physical orientation or scale. With this, the system successfully accommodates exploration of movement qualities displayed within an extensive video corpus. We design a task that allows participants to rank and rate the kinematic similarities of video segments retrieved using different feature sets. Within a limited number of examples, participants universally agreed that the content from neighbouring segments were more similar compared to those selected at random. Neighbouring segments taken form the same choreography, that were nearly identical received highly positive similarity scores. Where the average score for the other pairings were largely underwhelming, we consider improvements towards our feature representation and use of data. The feedback also concludes a superior effectiveness of using features from both our visual and biometric representations of the movement, which we explore further during development. In our final reflections, we contemplate on how this study could be matured to invoke semantic descriptions of video, revealing the emotional content portrayed within one's physical gesture.

The motivation of this project was to evaluate the feasibility of quantifying physical gestures from video footage, with prospects that this format was highly in-tune with how we perceive and empathize with qualities of human movement, as well as being an accessible modality for data collection and distribution. Once the framework was setup, we were able to compare and evaluate the influence of different dataset and feature configurations at our own convenience. We could make intuitive judgments towards our model from observational analysis before requiring any formal testing. Once satisfied with the output, we were able to broadcast a batch of results and obtain feedback fairly quickly. With that in mind, we encourage the use of video embeddings as part of the development process regarding future machine learning projects associated with movement qualities.

> "*I think the answer lies in having two screens, one of which there is a video of the dance, on the other there is a sort of notation that accompanies, moves along with the dance and is three-dimensional. It can be stick figures or whatever, but they move in space so you can see the details of the dance; and you can stop and slow it down. This can certainly be done, it would just take a lot of time and money! I once asked a computer specialist about it and he said, 'Oh it's perfectly possible, it would just take one month and one million dollars' (laughter)*" ***Merce Cunningham, The Dancer and the Dance p. 188 [17]***

7 ACKNOWLEDGEMENTS

All the data used in this project is intended for research purposes only

The LMA examples video dataset is provided by Leon Eckert, which includes demonstrations from Dominique Vannod

The Merce Cunningham Split Sides solo recordings are taken from *Merce Cunningham Dance Company: Split Sides by Merce Cunningham* from the Oxford University Bodleian Library video archives

The rest of the video data was taken from the *Prix de Lausanne* online media collection

8 SOURCE CODE & ONLINE RESOURCES

1) Project Source Code: http://gitlab.doc.gold.ac.uk/wprim001/Dance2Vec

   Includes Processing code, iPython notebook, utility scripts and example data

2) Interactive Visualizer Script: https://github.com/wprimett/Dance2Vec_Visualizer

   Includes html/JavaScript source with exported json files

3) Interactive Visualizers

   Main: http://igor.gold.ac.uk/~wprim001/MOCO/demo/demo.html

   Biometric Features Only: http://igor.gold.ac.uk/~wprim001/MOCO/demo/demoBio.html

4) Online User Test: https://patch-cupcake.glitch.me

5) Feature Visualization Examples: http://igor.gold.ac.uk/~wprim001/MOCO/final/

REFERENCES

[1] Cao, Z., Simon, T., Wei, S. E., & Sheikh, Y. (2017, July). Realtime multi-person 2d pose estimation using part affinity fields. In CVPR (Vol. 1, No. 2, p. 7).

[2] COCO - Common Objects in Context. Cocodataset.org. Retrieved 23 February 2018, from http://cocodataset.org

[3] Fdili Alaoui, Sarah, Jules Françoise, Thecla Schiphorst, Karen Studd, and Frederic Bevilacqua. "Seeing, Sensing and Recognizing Laban Movement Qualities." In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, pp. 4009-4020. ACM, 2017.

[4] Genetic Dance Algorithm. (2015). Genetic-dance-algorithm.leoneckert.com. Retrieved 23 February 2018, from http://genetic-dance-algorithm.leoneckert.com/

[5] OpenPose: Real-time multi-person keypoint detection library for body, face, and hands estimation. GitHub. Retrieved 23 February 2018, from https://github.com/CMU-Perceptual-Computing-Lab/openpose

[6] Sicchio, K. Layering the Choreographic Process: Making Dance Work with Machine Learning. Proceedings of MICI 2017: CHI Workshop on Mixed-Initiative Creative Interfaces., (2017).

[7] Wei, S. E., Ramakrishna, V., Kanade, T., & Sheikh, Y. (2016). Convolutional pose machines. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4724-4732).

[8] Home - Prix de Lausanne. Prix de Lausanne (2018). https://www.prixdelausanne.org/.

[9] Selections, 2014 - YouTube. YouTube, 2014. https://www.youtube.com/playlist?list=PLe146Eh6XHuSzfXdDwShLkiA7ByTONuwA.

[10] Dance Detail - Merce Cunningham Trust. Mercecunningham.org, 2018. https://www.mercecunningham.org/index.cfm/choreography/dancedetail/params/work_ID/171/.

[11] LeCun, Y., Cortes, C. and Burges, C. MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges. Yann.lecun.com, 1998. http://yann.lecun.com/exdb/mnist/

[12] Gillies, M., Brenton, H., Yee-King, M., Grimalt-Reynes, A. and d'Inverno, M. Sketches vs skeletons: video annotation can capture what motion capture cannot. Proceedings of the 2nd International Workshop on Movement and Computing - MOCO '15, (2015).

[13] Crnkovic-Friis, L., & Crnkovic-Friis, L. (2016). Generative choreography using deep learning. arXiv preprint arXiv:1605.06921.

[14] Turquois, C., Hermant, M., Gómez-Marín, D. and Jordà, S. 2016. Exploring the Benefits of 2D Visualizations for Drum Samples Retrieval. Proceedings of the 2016 ACM on Conference on Human Information Interaction and Retrieval - CHIIR '16.

[15] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).

[16] Alaoui, S., Caramiaux, B., Serrano, M. and Bevilacqua, F. 2012. Movement qualities as interaction modality. Proceedings of the Designing Interactive Systems Conference on - DIS '12.

[17] Cunningham, M. and Lesschaeve, J. 1999. The dancer and the dance. Marion Boyars, London.

[18] Chi, D., Costa, M., Zhao, L. and Badler, N. 2000. The EMOTE model for effort and shape. Proceedings of the 27th annual conference on Computer graphics and interactive techniques - SIGGRAPH '00.

[19] Merce Cunningham - Merce Cunningham Trust. 2018. Mercecunningham.org. https://www.mercecunningham.org/merce-cunningham/.

[20] Mancas, M., Frisson, C., Tilmanne, J., d'Alessandro, N., Barborka, P., Bayansar, F., ... & Laraba, S. (2015). Proceedings of eNTERFACE 2015 Workshop on Intelligent Interfaces. arXiv preprint arXiv:1801.06349.

[21] Müller, M. (2007). Information Retrieval for Music and Motion (pp. 235-236). Berlin, Heidelberg: Springer-VerlagBerlinHeidelberg..

[22] McInnes, L., & Healy, J. (1802). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv 1–18 (2018).

[23] Dalal, N. and Triggs, B. Histograms of Oriented Gradients for Human Detection. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05).

[24] Larboulette, C. and Gibet, S. 2015. A review of computable expressive descriptors of human motion. Proceedings of the 2nd International Workshop on Movement and Computing - MOCO '15.

[25] Expressive Movement Qualities for Embodied Interaction Design: A Brief Introduction. Retrieved 23 February 2018, from http://igor.gold.ac.uk/~wprim001/AdvancedTopics/Expressive_Movement_Re port.pdf

[26] Primett, W. and Tanaka, A. Automated Segmentation of Video Footage Using HighAccuracy Pose Estimation and Movement Qualities. Goldsmiths Computing. http://igor.gold.ac.uk/~wprim001/MOCO/MOCO_submission_Feb18.pdf.

[27] Centre of Mass Lab. 2018. University of Minnesota. https://www.d.umn.edu/~mlevy/CLASSES/ESAT3300/LABS/LAB8_COM/cm.htm.

[28] Winter, D. 2009. Biomechanics and motor control of human movement. Wiley, Hoboken, N.J.

[29] Fedden, L. 2017. Comparative Audio Analysis With Wavenet, MFCCs, UMAP, t-SNE and PCA. Medium. https://medium.com/@LeonFedden/comparative-audio-analysis-with-wavenet-mfccs-umap-t-sne-and-pca-cb8237bfce2f.

[30] Borg, M. 2016. Multi-Dimensional Reduction and Visualisation with t-SNE. Mark-borg.github.io. https://mark-borg.github.io/blog/2016/tsne/.

[31] McInnes, L. 2018. Uniform Manifold Approximation and Projection. GitHub. https://github.com/lmcinnes/umap.

[32] Dalal, N., Triggs, B. and Schmid, C. 2006. Human Detection Using Oriented Histograms of Flow and Appearance. Computer Vision – ECCV 2006, 428-441.

[33] Mouselimis, L. 2018. Image classification of the MNIST and CIFAR-10 data using KernelKnn and HOG (histogram of oriented gradients). Cran.r-project.org. https://cran.r-project.org/web/packages/KernelKnn/vignettes/image_classification_using_MNIST_CIFAR_data.html.