8-14-2017

Synthesizing Human Actions with Emotion

Mousumi Deb

SYNTHESIZING HUMAN ACTIONS WITH EMOTION

by

Mousumi Deb

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

Major: Electrical and Computer Engineering

The University of Memphis

December 2017

ACKNOWLEDGMENTS

I would like to express gratitude to my advisor, Dr. Bonny Banerjee for his support, advice, and encouragement during my research. It has been a great privilege to work with and learn from him.

I am grateful to the respected committee members, Dr. Aaron L. Robinson and Dr. Madhusudhanan Balasubramanian for their time and feedback regarding my thesis.

I would also like to thank Dr. Andrew Olney, Associate Professor, Institute for Intelligent Systems, for his help regarding the advice regarding dataset and technologies.

I am grateful to all my lab mates at the Computational Intelligence Laboratory (CIL).

Finally, I must express my very profound gratitude to my parents, husband Abhijit K Nag and sister for providing me with unfailing support and continuous encouragement throughout the study and research. This accomplishment would not have been possible without them.

ABSTRACT

Realistic synthesis of human actions is a challenging problem. This thesis investigates the problem of synthesizing actions, with individual variability, under different emotions. Current action/gesture synthesis, understanding and recognition models do not provide a general framework for synthesizing an extensive range of actions over a large range of emotions. The literature on spectral style transfer provides a plethora of viable approaches for transferring the style of action learned from one individual to another. Our idea is to consider an emotion as a style and then use a style transfer algorithm for transferring an emotion from one action to another. This allows us to synthesize any action over a large range of emotions. Experiments reported in this thesis are based on generating 18 actions with five emotions using the Kinect skeleton. The quality of the synthesized actions over time is evaluated through a subjective perception test, which is a standard in the domain of gesture synthesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Synthesizing human's everyday actions embedding emotion poses significant interest in various cross-disciplinary research domains. Human action and emotion are an integral part of social interaction and affecting social outcomes (Vosk, Forehand, & Figueroa, 1983). Emotions can be expressed through different modalities such as facial expression and physical action. Facial expressions have been the most extensively studied, whereas the physical actions are less studied (Lhommet & Marsella, 2014).

Motion capture equipment and Kinect can provide realistic and smooth skeleton points (Wang et al., 2014). However, these skeleton points capturing systems are expensive and time-consuming and also laborious to use. The automatic of generation the actions with emotions can overcome these issues.

1.1 Overview

Style transfer method allows transforming an input action into a new style action while preserving its original content (Hsu, Pulli, & Popović, 2005). The concept of spectral style transfer for synthesizing human motion enables to transfer style of the frame by frame spatial similarity between two independent actions (Yumer & Mitra, 2016). Our idea is to use a style transfer algorithm for transferring the style of action learned from one emotion to another. In this approach, we can learn important frequency component of target emotion based action from the database and apply to input. This approach allows us to synthesize any action over a large range of emotions. In this work, we utilized the CMU graphics (Hahne, 2010) and Emotional Body Motion Database (Max-Planck Institute for Biological Cybernetics in Tuebingen, 2014) human motion data preprocessing for skeleton compatibility. We extracted the difference of target emotion based source action and reference emotion based action skeleton in the spectral domain. Afterwards, we applied the difference to the input action and combined with the constant input phase. Then, we reconstruct the time domain action signal from the spectral domain. The

procedure of generation of actions with emotions in a block diagram is shown in Figure 1.1. In the motion data, energy is concentrated in upper or lower body based on the actions such as upper body actions (e.g., hitting) and lower body action (e.g., kicking). Considering the style transfer method is energy based, based on the upper or lower body input action we are applying the upper or lower training data similarities to do the best style transfer. Therefore, we have divided the training data upper body (e.g., hitting) and lower body (e.g., kicking). In this thesis, we have experimented 18 actions with five emotions using the skeleton joints. This work will help to build and enhance dataset actions with various emotions. Finally, we evaluate our generated actions with emotions via quantitative experiments. We conducted a user study on Amazon Mechanical Turk (AMT) to evaluate the synthesized actions with emotion video. In addition, we discuss our and state of the art evaluation result.

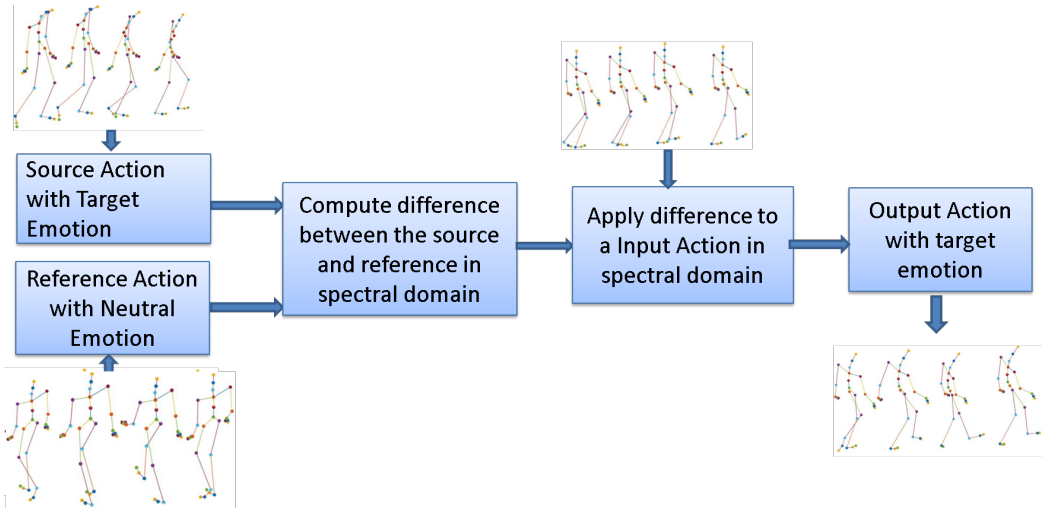


Fig. 1.1: Overview of generation actions with emotions

1.2 Contributions

The main contribution of this thesis work is:

1. An approach to synthesize any action over a large range of emotions.
2. Implementation and evaluation of the approach using subjective perception test.

1.3 Outline

The rest of the thesis is arranged as follows. A thorough review of the state-of-the-art is presented in Chapter 2. Following this, the motivation of using style transfer method to synthesize actions with emotion and the brief description of the method are given in Chapter 3. In chapter 4, dataset details and implementation results are discussed. The evaluation criteria and evaluation results are also presented in Chapter 4. Finally, in Chapter 5, we discuss the conclusion and future work of synthesizing actions with emotion.

Chapter 2

Related Work

Our work is based on the literature on spectral style transfer for synthesizing human motion. In this chapter, a brief overview of related work of human action synthesis is presented and different kind of motion capture technologies is discussed. In the end, a description of BVH(Bivison) file is presented.

2.1 Motion Synthesis Methods

According to the literature, existing motion synthesis methods can be classified mainly into following categories: manual synthesis, physics-based methods and data-driven methods (Wang et al., 2014).

Manual synthesis is the earliest and most basic motion synthesis approach which refers manually setting degrees of freedoms(DOFs)of human joints in all individual points in times which is called key frames. The manual synthesis was majorly applied in early cartoon movie and game industries (Wang et al., 2014). Different kinds of interpolation methods, such as cubic spline interpolation are used to compute DOFs in between the key frames. Perlin and Goldberg (1996) have demonstrated manual synthesis computationally efficient algorithm based on plain interpolation. Though simple interpolation hardly produces realistic outputs unless the key frames are remarkably dense, the artist should manually construct a substantial fraction of the character poses that used to appear in the motion. Moreover, this work is time-consuming because a lot of key frames must be created, and it is challenging to draw right key frames which are realistic and smooth when played in sequence. Besides, it requires some knowledge of art sometimes. Mostly the result of manual synthesis is very less natural and the simulation depends on enormous manually modeling work.

The concept of physics-based methods depends on the real human movements likewise the physical law. These methods are used to appear in both graphics and robotics (Fang & Pollard, 2003). In physics based method, the mass distribution for each body part

and the torques generated by each joint and Newton's laws serve a system of ordinary differential equations(ODEs). ODE that can be integrated to yield joint trajectories and the trajectory of each joint can be retrieved by solving the ODE. This approach eliminates a great deal of manual labor by, using physical laws to automatically fill in the details of the motion of a character and guarantees that motions are physically accurate. However, the physical accuracy does not imply visual realism. Besides, the main disadvantage of the physical based methods is difficult to design specific equation of motion. Moreover, the generated movement has fewer details and lack of individuality.

The data-driven methods allow reuse of existing data to synthesis new data. In data driven approach, original or raw material data can be captured by the motion or skeleton point capture equipment which will contain a source of highly realistic example motions. Moreover, by using this approach the result can be used as the source of the human motion data. Data-driven methods have the advantages of producing natural, realistic and more generative motions in comparison with Physics-based motion synthesis. Also, there is no need to build complex control systems for each joint (Wang et al., 2014). In the literature, most of the algorithms have been proposed to synthesize new motion based on motion capture data. In this part, we present the brief description of the state of arts of data-driven motion synthesis algorithms.

Motion blending is the way to generate a new motion from pre-recorded motion and concatenate two motion and blend the transition part. Perlin (1995) demonstrated motion blending algorithm which are based on the real-time procedural animation system. To synthesize a new motion using motion blending process, a user manually constructed a set of base motions and then used blending operations via interpolation to transition between motions (Kovar & Gleicher, 2003). Radial basis functions (Wooten & Hodgins, 2000) can be used to blend two motion if the motion capture data is contained by relative orientation angles. Kovar and Gleicher (2003) introduced the registration curves for computing blends which support all the operations such as interpolation, transitioning,

and continuous control and could be used as a back end in these systems. Sometimes for synthesizing some special motions, it is hard to find suitable matching motion capture. For that reason, it is always required to capture new data for this purpose. Therefore, only blending and interpolation cannot make use of motion capture data efficiently and perfectly.

Motion graphs generate continuous streams of motion under user specified constraints not only for concatenate two motion samples as motion blending. This approach is sometimes called ”move trees”, and it has been used for a long time in the video game industry for the character control (Mizuguchi, Buchanan, & Calvert, 2001). This approach is more flexible and can make use of motion capture data more efficient than simply blending. In addition, using motion graph approach is also feasible to combine environment constrains or user specified constraints, for instance, sneak walking in this approach. Most of the motions more than a few seconds in duration are naturally thought of as sequences of atomic actions. These motion graphs have historically been constructed manually in the sense that a user explicitly decided which motion clips could be connected. In another approach (Kovar, Gleicher, & Pighin, 2002), a simple linear blending method to concatenate two motions clips and showed that a motion graph could be automatically constructed by identifying places where motions are locally similar. Kovar et al. generated a transition from the ith frame of the first motion to the jth frame of the second motion by linearly interpolating the root positions, performing spherical linear interpolation on joint rotations, and putting additional constraints on the desired motion (Wang et al., 2014). However, the limitation of the motion graph is generating motion at the transition point where the thresholds for similarity must be specified by hand. Because different kinds of motion have different fidelity requirements.

In the style transfer process, input movement style can be transformed into a new style at the same time keeping its own original content. Real-time data driven method using linear time invariant model are used to encode style differences and variances (Hsu

et al., 2005). However, the performance of the method degrades when the motion sequences are complex (Hsu et al., 2005). Recently a data driven method (Xia, Wang, Chai, & Hodgins, 2015) has circumvented from (Hsu et al., 2005) linear time invariant method to a local mixture of autoregressive method for style transfer using a motion dataset. In this method, motion sequences labeled in various styles and actions and a local mixtures of an autoregressive models creates temporally local nearest neighbor mixtures from the source style database to transform each frame successively (Xia et al., 2015). In other words, this method is able to create optimal stylize every frame in spatio-temporal space. This method able to handle more heterogeneous action motion compared to other methods. Moreover, this method generates better result when the style dataset contains the type of actions in the target motion sequences. In one of the previous work stylistics motion generation method (Amaya, Bruderlin, & Calvert, 1996), using the signal processing method emotional transformed has been calculated then those transformation applied to existing motions of articulated figures to generate the similar motions. To calculate the transformation (Amaya et al., 1996) they have calculated the difference between neutral and emotional movement.After that, the difference has been applied to a new neutral movement. Moreover, Ikemoto, Akrikan, and Forsyth (2009) introduced motion modeling method using gaussian process models to demonstrate the stylistic motion.

Statistical methods of input output relationships are a primitive tool in closely all fields of science engineering. The statistical models are mostly described as a set of mathematical equations or functions that defined human motion using a finite number of parameters and their associated probability distribution. In the translation of styles of line drawings (Freeman, Tenenbaum, & Pasztor, 2003) the statistical methods have been used. For synthesizing the stylistical motion, the generative statistical method such as Hidden Markov Models (HMM) was used (Brand & Hertzmann, 2000). Statistical methods, for instance, the Gaussian Process dynamic model (GPDM) was applied in J.M. Wang, Fleet

and Hertzmann (2005). After that, J. M Wang et al.(2007) extended GPDM to Gaussian Process Latent Variable model (GPLVM) to capture stylistic variation caused by gait and identity. The capturing the same style and identity in the same action method has been extended later (Min, Liu, & Chai, 2010).

Deep learning framework is used to synthesize character movements based on high-level parameters, and the system can produce smooth, high-quality motion sequences without any manual pre-processing of the training data (Holden, Saito, & Komura, 2016). However, there is some ambiguity present in the output data. Besides, a good amount of data is required to train a network to produce the data using the deep learning framework (Holden et al., 2016).

2.2 Motion Capture

The process of recording and registering the movement of objects or human is defined as Motion capture (mocap). The mocap technology is a successful technique for generating realistic animations. In addition, it is also the foundation for motion data-driven motion synthesis approaches and it has great use in the entertainment industry for films and games to get more realistic human movements. Figure 2.1 displays different types of motion capture technologies as well as some of the motion capture technologies processes.

2.2.1 Motion Capture Technologies

Capturing the movement of the human body using electromagnetic sensors is called magnetic motion capture system. In a magnetic motion capture system, electromagnetic sensors are cabled to an electronic control unit which correlates their reported locations within the field. In other words, these sensors are connected to one or multiple computers which are able to process the data and produce 3D data in real time. The generated 3D data in real time which represent positions in 3D space tracking trajectory will be display on the screen at the same time. Moreover, the generated output is accurate and fast, without complex post-processing time. However, the main drawbacks of magnetic motion capture are heavier sponsors and restricted freedom of movement

caused by the connected cables (Lonkar, 2017). In addition, there are also some other problems like magnetic distortion happens as distance increases, prone to interference from magnetic fields and the system is very expensive (Lonkar, 2017).

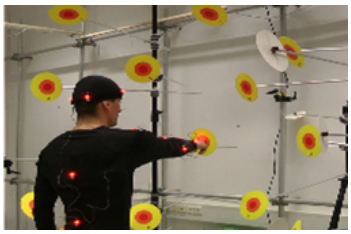
In electro-mechanical systems, participants need to wear this special suit with integrated electro-mechanical sensors that register the motion of the different articulations that is hooked on to performers back. In this process, each joint has sensors which provide the position (Lonkar, 2017). The main advantages of the electro-mechanical systems are high tracking quality and real-time processing. However, it reduces the cost and constraints of movement because of cabling. The electro-mechanical sensor suit is one of the common examples. The disadvantages of this system are that it is not able to track the background of the actor, and the tracking skeleton is usually constant (Du, 2014).

Depth cameras have become very popular in recent years. A good example of depth camera motion capture system is Kinect camera. For motion tracking, the depth information is a very important feature, and it also tracks the background of the actor. Thus, it has good interaction property. The depth cameras, such as Kinect, are cheaper than these special sensors. In addition, it also has the benefits, for instance, full freedom of movement and the interaction between multi-actors. However, currently, the raw depth data from Kinect is quite noisy. In addition, trained actor or subjects are required to synthesis the data. Therefore, collecting data using Kinect is costly and time-consuming

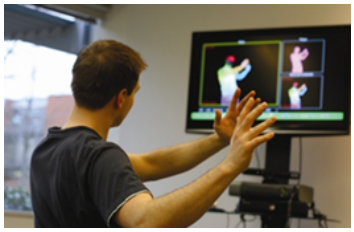
Optical motion capture systems used to employ proprietary video cameras to track the motion of reflective markers (or pulsed LED's) connected to particular locations of the actor's body (*Optical Motion Capture Systems*, 2017). Optical systems have been applied to motion capture area based on photogrammetric methods. Traditional optical camera systems provide the advantage of full freedom of movement and interaction with different actors as like as the depth camera system. Optical Motion Capture system is mostly very flexible. Therefore, it can be extended by adding multiple cameras, which could give larger tracking area and better tracking results. The optical camera tracking system can be

divided into maker based and makerless. The captured motion data is usually 3D position and relative orientation of each marker or joint in human skeleton both in marker based and markerless. The main drawback of the optical systems is post-processing is necessary to extract captured motion data from record videos.

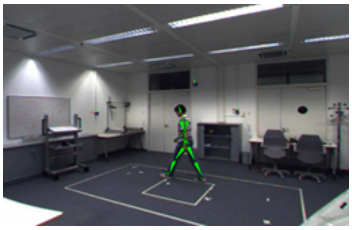
(a)



(b)



(c)



(d)

Fig. 2.1: Motion Capture Technologies (Du, 2014): (a) Magnetic motion capture (b) Electro-Mechanical Motion capture (c) Depth Camera Motion capture (d) Optical Motion Capture

Animation software is one of the popular technologies used to create realistic action based animation. In this thesis, we experimented with some of the animation software. One of them makes human, which is an open source 3D animation tool (Team, n.d.). Besides, we have experimented on iPi recorder animation software to make an own 3D character (Starbuck, Seo, Han, & Lee, 2014). However, we have faced some problems, such as the iPi recorder produces only body movement excluding hands movement. It is hard to know whether synthesized data will generalize to the real world. Moreover, in some animation software hardware equipment also required recording the data. Therefore, this software was not efficient for our work.

2.3 Motion Data Representation

Motion Data is defined as the time series data where each frame represents the poses of the character and poses of the character could be different. In the motion data format, the poses of the character are typically parameterized by the character joints positions or angles (3 dimensional).

2.3.1 Motion Capture File Formats

In Table 2.1 the list of motion capture file formats and references along with motion capture data formats in use today along with URLs for additional formatting information are given according to Meredith et al. (2001). The storage format of the motion capture data is different according to different manufactures listed in Table 2.1. In our work, we experimented in BVH format. The general skeleton structure (Wang et al., 2014) shown in figure 2.2 (a) is used to define the skeleton joint chain, where each joint is connected based on hierarchical structure. The hierarchical structure of the skeleton is shown in Figure 2.2(b).

Table 2.1: List of Motion capture file formats and corresponding references for additional format information (Meredith et al., 2001).

Mocap Extension	**Associated Institution**	**File Format Reference**
ASF and AMC	Acclaim	http://www.darwin3d.com/gamedev/
BVA and BVH	Biovision	https://research.cs.wisc.edu/
C3D	The biomechanics standard	https://www.c3d.org/
ASK/SDL	Biovision/Alias	http://research.cs.wisc.edu/
CSM	3D Studio Max, Character Studio	http://www.character-studio.net/
HTR and GTR	Motion Analysis software	https://research.cs.wisc.edu/graphics/
BRD	LambSoft Magnetic Format	http://www.dcs.shef.ac.uk/
TRC	Opensim Motion Analysis	http://simtk-confluence.stanford.edu/
MOT and SKL	Acclaim-Motion Analysis	http://www.cs.wisc.edu/
MNM	Autodesk 3D Studio Max	http://autodesk.com

BVH File Format

The BioVisions BVH file contains two parts, the first part represents the hierarchy and initial pose of the skeleton and the second part refers to the motion data section, which describes channel data referred in the first part of each frame. The human skeleton is organized in a tree structure, normally starting from the root node in BVH file shown in Figure 2.2(b). Recently, BVH format has become a recursive definition. Each segment of the hierarchy includes some data relevant to just that segment then it recursively defines its children. The main keywords of the BVH format are given below:

- **Hierarchy:** The BVH file starts with HIERARCHY keyword, which defines the actors skeleton in a hierarchical structure. It allows to define more than one actors skeleton in Hierarchy section, and the hierarchical skeleton usually starts from root joint.

- **Offset:** Offset defines a 3D vector, which contains length and direction used for drawing the parent. In addition, Offset also represents the bone length between two joints.

- **Channels:** Channels represents the degree of freedom of a human skeleton. Each bone within a skeleton can be subject to position, orientation and scale change over the time of the animation, where each parameter is referred to as a channel.

- **Motion:** The motion capture data at each timestamp contains in the ending part of BVH file. In the file, each line of motion represents each frame, and the data is ordered according to sequentially concatenating the channels in hierarchy part.

The BVH file defines a human skeleton in it, so we do not need to pre-define a human skeleton for different motion capture system, which makes our statistical modeling approach more robust for different human skeleton model. To calculate the global position of each joint in the human skeleton from raw BVH data, the mathematical definition was

introduced by Meredith et al. (2001). A BVH data file has the absolute translation and rotation of the root joint and relative rotations of all other joints. Skeleton position of each joint is crucial for our action synthesis work. Skeleton Joint, $3 \times T$ matrix, where T is the total number of frames in our training data.

The Kinect skeleton joint of the bvh data format with 38 joints displays in Figure 2.2(a). Each joint has three channels or degrees of freedom x, y, z coordinates. The hierarchical definition of the joints are shown in Figure 2.2(b).

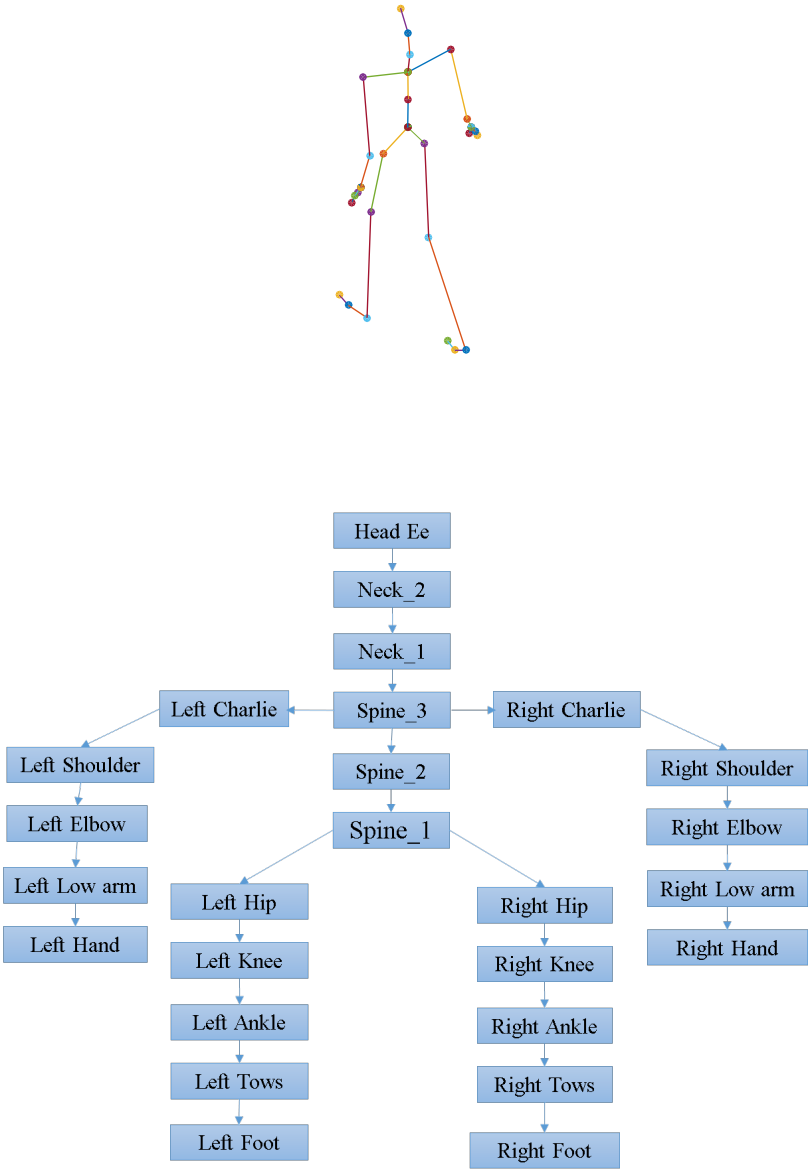


Fig. 2.2: Human skeleton structure: (a) General human skeleton structure with 38 joint points (b) Hierarchical structure of the BioVisions data format (Wang et al., 2014)

Chapter 3

Methodology

In this thesis, for synthesizing the actions with emotions, we are using the literature on spectral style transfer. Style transfer method is data-driven which allows utilizing the prior knowledge embedded in prerecorded skeleton data for the style transfer (Xia et al., 2015). In this section, the description of the core methods of spectral style transfer is presented.

3.1 Introduction

The human motion skeleton points presented as the time domain signal is shown in Figure 3.1. In this figure, we can observe that the specific actions (for example walking and kicking) with different emotion (for example neutral and kicking) in time domain there is no trivial correlation that can be utilized immediately. Therefore, style transfer in time domain signal is quite complex because there is no existing significant correlation, especially when the actions are different in Figure 3.1. On the other hand, these same data in magnitude component in the spectral domain as shown in Figure 3.2. We utilized that the correlation is easily computable and also comparable to each other. Thus, these signal frequency components in spectral domain contain useful information (Yumer & Mitra, 2016). The difference of neutral and emotion based skeleton joints is highly correlated in frequency domain magnitude component even when the actions are different shown in Figure 3.2. These experiments deduce the fact that if we have reference actions data performed in the different emotion, we can extract the difference and apply the difference to a new action data.

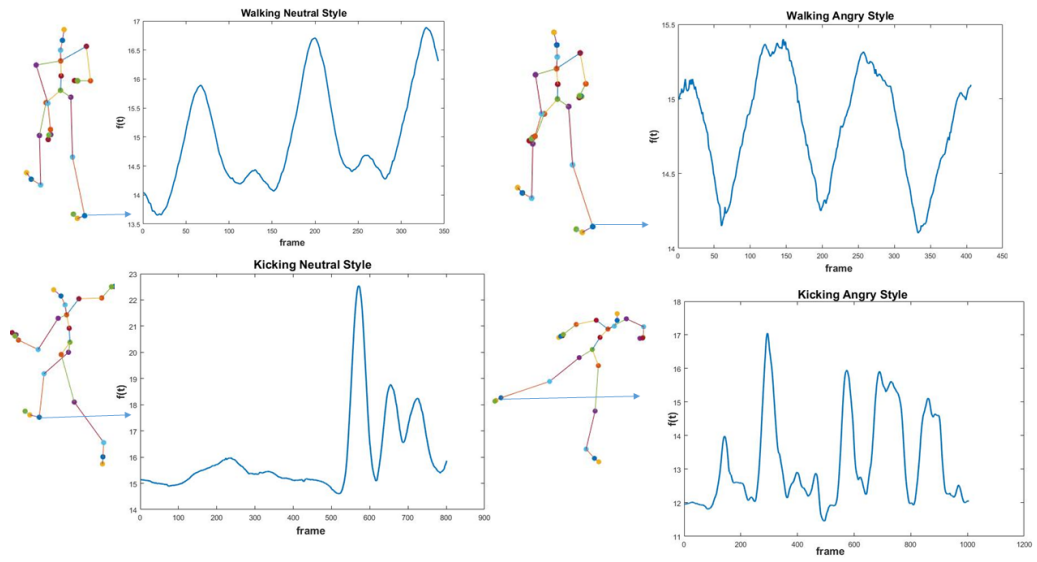


Fig. 3.1: The difference of neutral and stylized skeleton joints. Where f(t) Discrete Time domain signal of one of the degrees of the freedom (DOF) of the skeleton joints

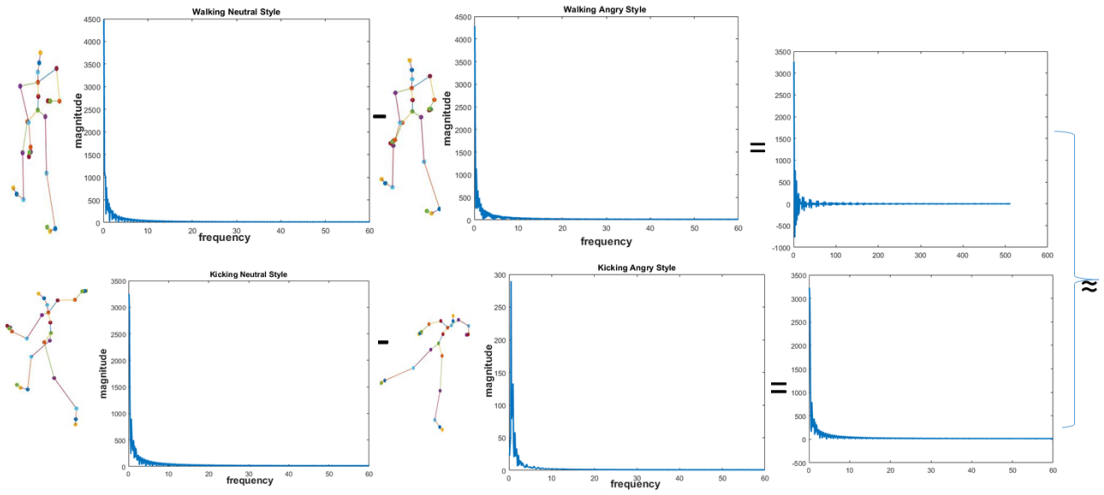


Fig. 3.2: Frequency domain representation of (top) walking and kicking (bottom). The difference of neutral and stylized skeleton joints are highly correlated in frequency domain magnitude component even when two actions are different.

3.2 Method Definitions

We now define the terms and concepts relevant to Style Transfer algorithm.

Discrete Fourier transform: Let $f[t]$ be a discrete time domain signal of one of the

degrees of freedom (DOF)of a feature. Consequently, the Discrete Fourier Transform $F[k]$ of $f[t]$ is given by (Oppenheim & SCHAFER, 2009),

$$F[k] = \sum_{t=0}^{N-1} f[t] e^{-i} \frac{2 * \pi}{N} kt \quad k = 0, ..., N-1 \tag{3.1}$$

Where N is the length of feature and $i^2 = -1$. The single-sided spectrum $F[w]$ is given by

$$F[w] = \frac{2}{N} F[k] \quad k = 0, ..., N/2 \tag{3.2}$$

Where $w = (\frac{f_s}{N})k$ frequency transform from samples k in the time space. Here, f_s is the sampling frequency or the time difference between one frame to another frame of the original time signal $f[t]$.

Magnitude and Phase angle: Let, $R[w]$ and $A[w]$ is the magnitude and phase angle. Mathematically, $R[w]$ amplitude of combined cosine and sine and $A[w]$ is phase relative proportions of sine and cosine. The magnitude, $R[w]$, defines the existence and intensity of a motion at w frequency whereas the phase, $A[w]$ presents relative timing. The magnitude $R[w]$ and phase angle $A[w]$ follows:

$$R[w] = \big|F[w]\big| \tag{3.3}$$

$$A[w] = \angle F[w] \tag{3.4}$$

In this work, magnitudes encode whether there are specific the magnitude of the action and the angle represents the relative dynamic information of the skeleton data.

Inverse Discrete Fourier transform: Finally, to reconstruct $f[t]$ from $F[K]$ the inverse discrete transform (Oppenheim & SCHAFER, 2009):

$$f^{'}[t] = \frac{1}{N} \sum_{k=0} F^{'}[k] e^{i} \frac{2 * \pi}{N} kt \tag{3.5}$$

3.3 Algorithm on Style Transfer in Spectral Space

In this section, we describe the algorithm on Style Transfer in Spectral space. Let,

$f(t)$= time domain signal of the input action.

$f^s(t)$= time domain signal of the source style or target emotion.

$f^r(t)$= time domain signal of the reference style.

$f^s(t)$ and $f^r(t)$ from same action class.

$f^r(t)$ and $f(t)$ are from the same style from action class.

$f(t)$, $f^s(t)$ and $f^r(t)$ are computed discrete time signal of one of the degrees of the freedom(DOF). The main goal of Style Transfer in Spectral space model (Yumer & Mitra, 2016) is to extract the difference between the source style $f^s(t)$ and reference $f^r(t)$ in spectral space. Afterwards, we apply that difference into $f(t)$. The length of the three signal, synchronization and spatial correspondences could be different. Here, $R[w]$, $R^s[w]$ and $R^r[w]$ respectively, input, source, reference style spectral magnitudes. $R[w]$, $R^s[w]$ and $R^r[w]$ are computed using Eq.3.3 for all degrees of freedom in skeleton joints. The style transfer is formulated by applying the difference of $R^s[w]$ and $R^r[w]$ to $R[w]$ and computes a newly stylized magnitude $R'[w]$. To preserve the timing with first constraint and synchronization of the skeleton joints, the phase of the signal are constant (Yumer & Mitra, 2016) in the algorithm. We apply the magnitude difference of the resulting new magnitude $R'[w]$ is used in the inverse fourier transform presented in Eq. 3.5. We reconstruct the time domain data from frequency domain in Eq. 3.5. In addition, in the resulting signal in the method are real valued in the spatial-temporal space. The procedure of style transfer in spectral Space is summarized in algorithm 1. In this algorithm, for each 38 skeleton joint and DOF, we are calculating all the definitions of input, source and reference. Besides, we calculate the same magnitude component point in this algorithm for each action for mathematical calculation because input, source and reference action data might have the different number of frames.

Algorithm 1 Style Transfer in Spectral Space Pseudo code

Inputs: input signal $f(t)$, source signal $f^s(t)$, reference signal $f^r(t)$

Output: Output styled signal $f^{'}[t]$

for joint=1 to 38 **do**

  for DOF=1 to 3 **do**

    for frame=1 to n **do**

      Compute fft $F[k], F^s[k], F^r[k]$ by solving Eq. (3.1) and Eq. (3.2)

      Compute $R[w], R^s[w], R^r[w]$ by solving Eq. (3.3)

      Compute phase angle $A[w]$,$A^s[w]$and $A^r[w]$ by solving Eq. (3.4)

      Compute Difference between $R^s[w]$ and $R^r[w]$

$$R^s[w] - R^r[w]$$

      Apply the difference to input magnitude

$$R^{'}[w] = R[w] + (R^s[w] - R^r[w])$$

      Compute $F'[k]$by Adding $R^{'}[w]$ with input phase angle

$$F'[k] = R^{'}[w] + A[w]$$

      Compute ifft or reconstructed signal $f'[t]$ by solving Eq. (3.5)

    end for

  end for

end for

Output $f'[t]$

3.4 Schematic representation of Style Transfer in Spectral Space

The schematic representation of the algorithm style transfer is presented in Figure 3.3. Here source and reference data are available from the training database. In Figure 3.3, we are presenting the way of computing the difference between source and reference

action data. Afterwards, the resulting difference applied into the input magnitude component. In the end, we are combining the original phase information to new stylized magnitude component. Then, we reconstruct the new emotion based data in the time domain.

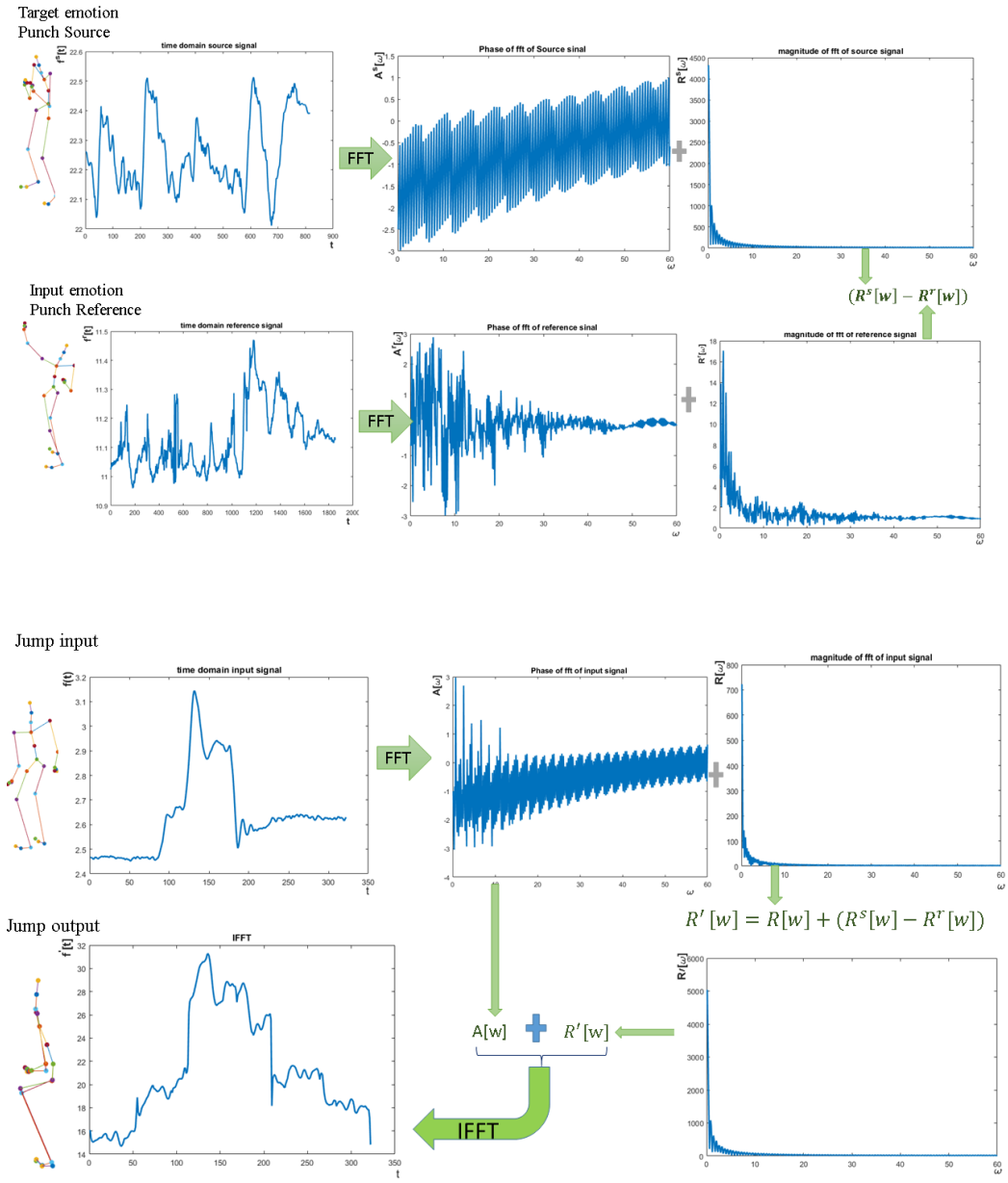


Fig. 3.3: Schematic representation of Style Transfer in Spectral space. (a) Compute the difference between $R^s[w]$ and $R^r[w]$. (b) Applying the difference to the input $R[w]$ and compute newly stylized magnitude $R'[w]$. $A[w]$ is kept constant while generating stylized time domain data.

Chapter 4

Implementation and Evaluation

In this chapter, dataset description, experimental setup and experimental results are discussed.

4.1 Dataset

In this work, we experiment on the two benchmark dataset CMU graphics lab (Hahne, 2010) and Emotional Body Motion Database (Max-Planck Institute for Biological Cybernetics in Tuebingen, 2014). The experimental data has 38 skeleton joints with three degrees of freedom for each joint.

4.2 Experimental Setup

We have tested on CMU motion capture database in BVH format (Hahne, 2010). The BVH format succeeded BioVisions BVA data format with the important addition of a hierarchical data structure describing the bones of the skeleton. The BVH file has two parts where the first part contains the hierarchy and initial pose of the skeleton, and the second part is the motion data section which describes channel data defined in the first part of each frame (*Biovision BVH*, n.d.). In BVH file, the human skeleton is represented in a tree structure, usually starting from the root node. Human skeleton can be reconstructed by deep first parsing this part. We have extracted the skeleton joints from the BVH motion data. In the data, skeleton joints were recorded in the motion data at 120 frames per second and mapped to 38 skeleton joints times of 3 degrees of freedom (Hahne, 2010). In the motion data, energy is concentrated in upper or lower body based on the actions such as upper body actions (e.g., hitting) and lower body action(e.g., kicking). Considering the style transfer method is energy based, based on the upper or lower body input action we are applying the upper or lower training data similarities to do the best style transfer. Therefore, we have divided the training data upper body action and lower body action is shown in Table 4.1.

Table 4.1: List of Actions Upper body and Lower body class actions

Upper body	**Lower body**	**Upper and Lower body**
Punch	Jump	Walking
Throwing	Sitting	Standing
Drinking	Falling	Running
Dressing	Reading	
Grabbing	Lying or Sleeping	
Pushing	Talking	
Cutting	Kicking	
Eating		

4.3 Experimental Results and Discussion

In this work, we generate 18 actions with five emotions of human skeleton joints by implementing the algorithm 1. The list of the actions is: walking, running, standing, punching, kicking, jumping, throwing, grabbing, hitting, sitting, lying, eating, dressing, drinking, falling, cutting, talking and pushing. The list of the emotions is neutral, happy, sad, angry and fear. We transform the input action skeleton pose data into a sequence of frames in the output emotion style on the fly. For expressing emotion through arm movement and gesture, an observation was made Gross, Crane, and Fredrickson (2001) is summarized in Table 4.2. We justify the emotion based action visually based on these features. Figure 4.1 shows sample key frames of input neutral walking and output sad walking, happy walking, angry walking and fear walking. The training data both source and reference for the ”sad walking” contained running data. According to the sad emotion features are shown in the Table 4.2, we can observe in output sad walking that skeleton head joints have the longest movement over time, the smallest amplitude of elbow motion and also hiding gesture. In Figure 4.2, sample key frames of input neutral running and output happy running, sad running, angry running and fear running and in this case

training data (source and reference) contained the walking data. We observe in the output that arms stretched out to the front according to the happy emotion feature Table 4.2.

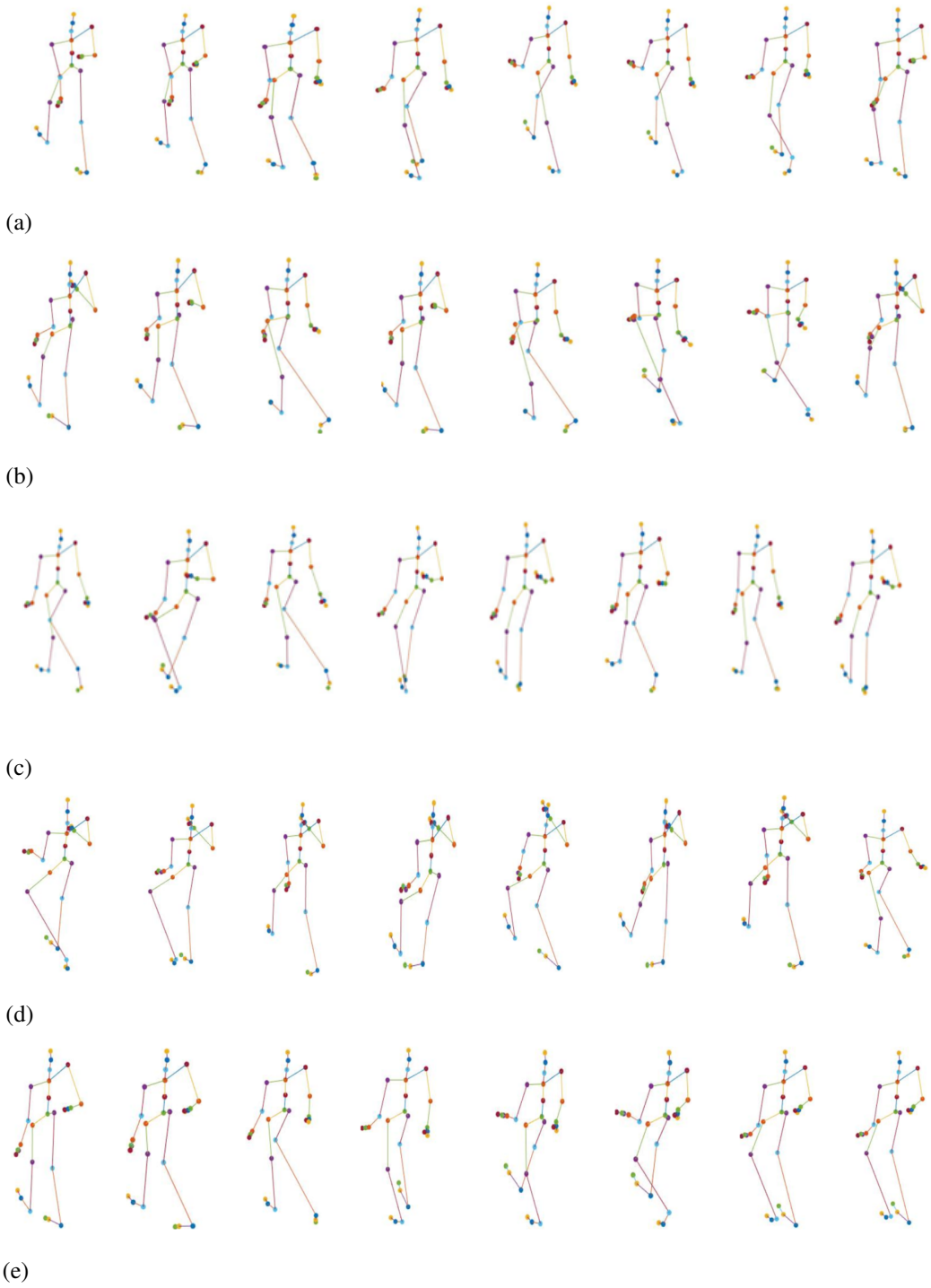


Fig. 4.1: Sample key frames of experiment (a) Input Neutral walk (b) Output Happy Walk (c) Output Sad walk (d) Output Angry Walk (e) Output Fear Walk.

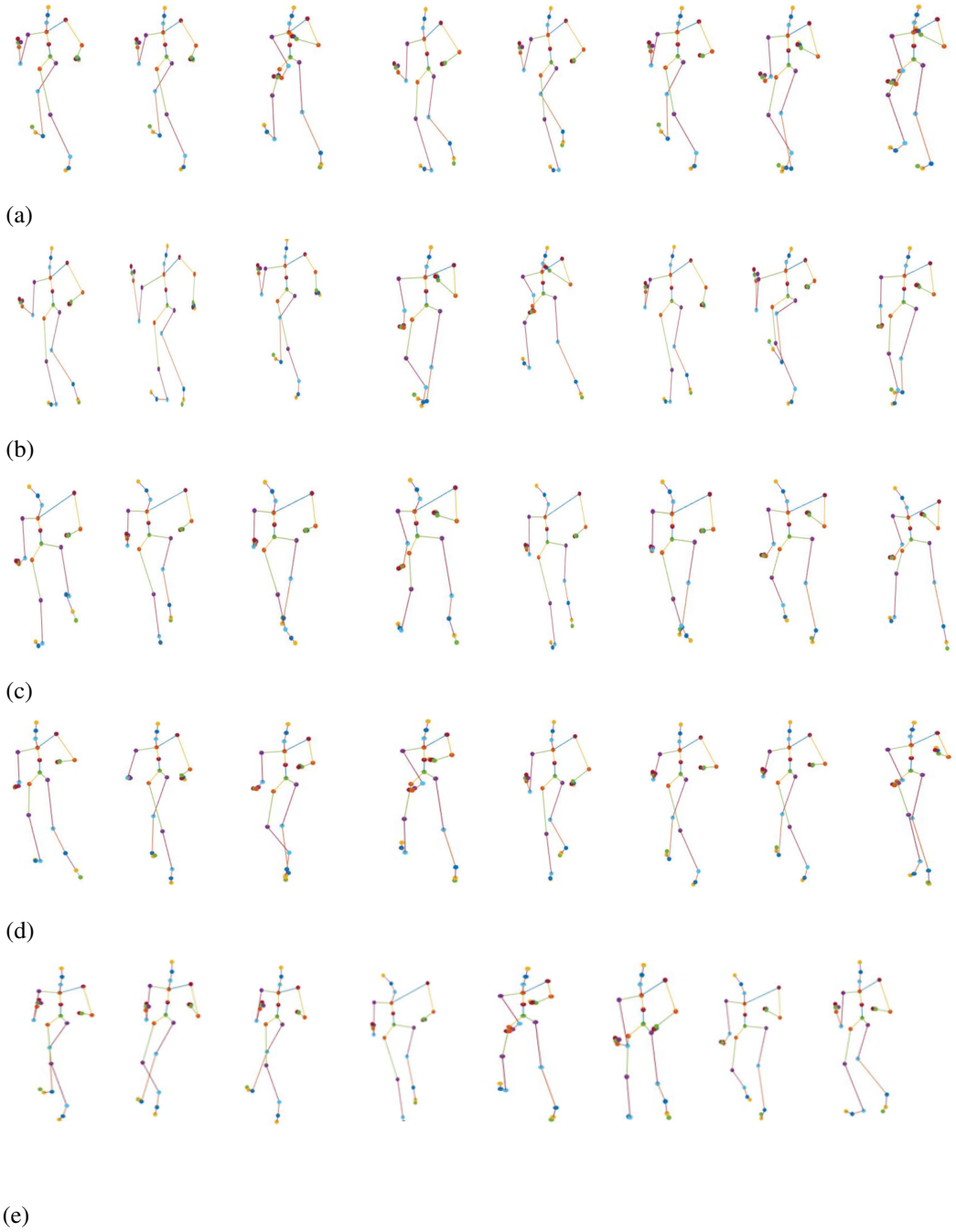
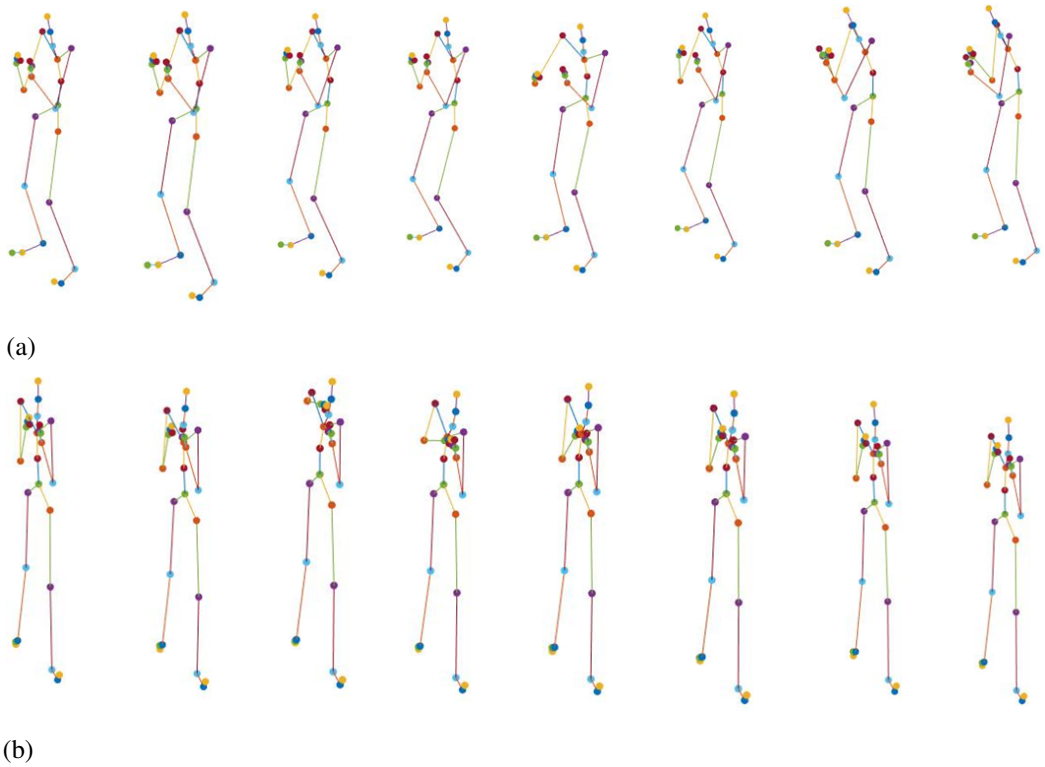


Fig. 4.2: Sample key frames of experiment (a) Input Neutral Running (b) Output Happy Running (c) Output Sad Running (d) Output Angry Running (e) Output Fear Running.

In Figure 4.3 sample key frames of input neutral punch and output angry punch when training data (source and reference) is standing. Based on the features of angry, we

can observe that liberalized hand-arm movement and arms stretched in towards the body, that is shown in Figure. 4.3(b)

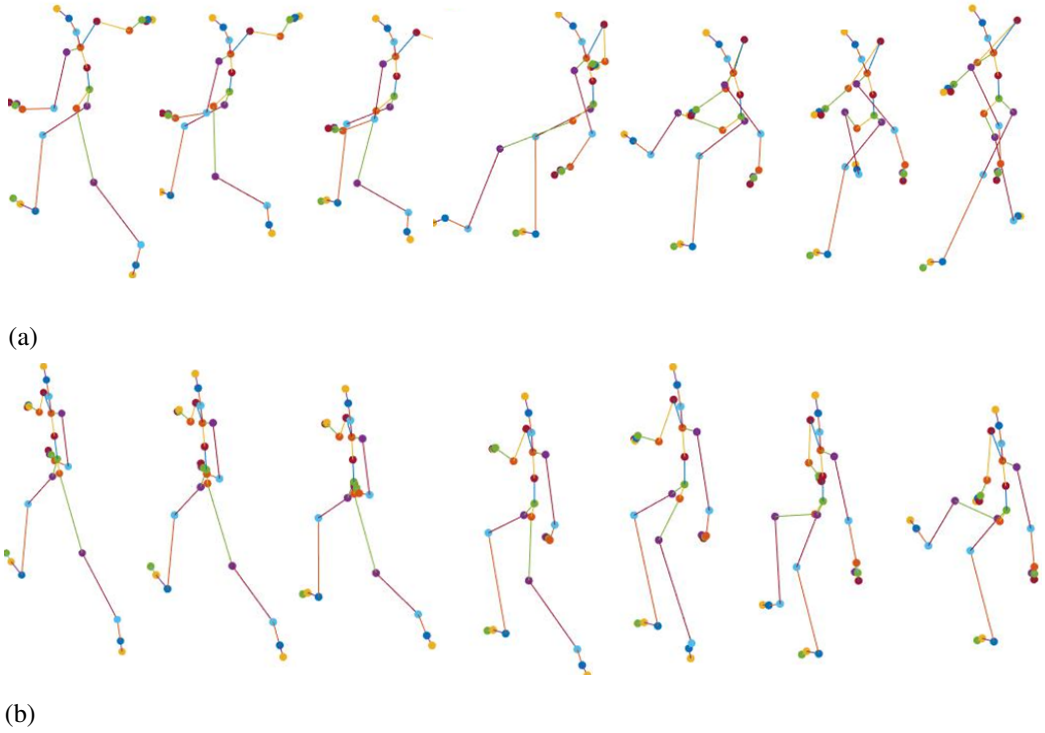


(a)

(b)

Fig. 4.3: Sample key frames of experiment (a) Input neutral punch (b) Output angry punch.

In Figure 4.3 sample key frames of input neutral kick and output angry kick when training data (source and reference) is standing. In this figure, the output Fear kick justifies the features of Table 4.2.



(a)

(b)

Fig. 4.4: Sample key frames of experiment (a) Input neutral kick (b) Output fear kick.

The overall computation time of the synthesizing actions with emotions is 32.40s.

Table 4.2: Feature of arm movement and gesture on emotion (Wang et al., 2014)

Emotion	**Features**
Angry	Lateralized hand-arm movement, arms stretched in, largest amplitude of elbow motion, largest elbow extensor velocity, highest rising arm.
Happy	Arms stretched out to the front.
Sad	Longest movement time, the smallest amplitude of elbow motion, least elbow extensor velocity and hiding and withdrawal gestures.
Fear	Arms stretched sideways

4.4 Evaluation Methods

In this section, we presented the brief description of the evaluation process of the thesis work. A user study has been conducted on Amazon Mechanical Turk (AMT) to evaluate the effectiveness of the proposed work.

Two different question types are designed and administered to assess both action and emotion from the generated video outputs. In the first question type, subjects are asked to choose the correct action and in the second question type, they are asked to choose the correct emotion. In this work, five survey forms are created where each form contains 30 videos of 18 actions and 5 emotions and a total of 60 questions. The collected responses are analyzed later to calculate the accuracy of the human actions and emotion of the proposed approach. The responses of the users are analyzed to ensure the quality of the responses to choose the best possible responses. For example, individual responses where the user recorded their response without waiting for the videos to finish are automatically rejected. Similarly, if all the responses of a subject are same for consecutive questions, they are also rejected as they do not reflect the actual attempt. In total, 50 responses per survey form have been used in this work to measure the efficiency of the proposed work.

4.5 Evaluation Results

The subjects are given each video of emotion based action and are asked to select the best action and emotion. The subjects were given 18 actions and five emotions along with "I have no idea" option. The user selected the best action and emotion from the options. We extract all the responses of the subjects for each action and emotion class C_i.We calculate the recognition rate using Eq. 4.1.

$$\textbf{Recognition rate} = \frac{\text{Number correct response for class}C_i}{\text{Number of Response for class}C_i} \tag{4.1}$$

The recognition rate of actions from the user study along with the recognition rate calculated using Eq. 4.1 and the recognition percentage value are listed in Table 4.3.

Table 4.3: Accuracy of action recognition from user responses

Action	**Accuracy**
Walking	97.78%
Standing	87.76 %
Running	87.20%
Jumping	87.80%
Punching	68.13%
Kicking	89.58%
Sitting	76.73%
Throwing	67.82%
Hitting	45.71%
Grabbing	50%
Pushing	39.29%
Drinking	60.34%
Dressing	45.450%
Eating	50%
Lying	83%
Talking	40%
Falling	51.85%
Cutting	40%

The recognition rate of emotions from the user study is shown in Table 4.4. Here recognition rate calculated using Eq. 4.1 and the recognition rate percentage value are given in Table 4.4.

Table 4.4: Accuracy of emotion recognition from user response

Emotion	**Accuracy**
Neutral	80.3383%
Happy	46.4706%
Sad	40.4444%
Angry	50.6579%
Fear	36.7347%

The confusion table for experimental output video of 18 actions are shown in Table 4.5, and five emotions are shown in Table 4.5.

Table 4.5: Emotion Recognition user-study confusion matrices. In each cell: emotion recognition percentage values are given.

		User Selected Emotion					
		Neutral	Happy	Sad	Angry	Fear	No Idea
Corrected Emotion	Neutral	80.3383	2.7484	2.1142	-	-	14.7992
	Happy	36.1765	46.4706	2.3529	2.9412	0.5882	11.4706
	Sad	40.8889	-	40.4444	-	7.1111	11.5556
	Angry	28.9474	5.2632	-	50.6579	5.9211	9.2105
	Fear	28.5714	-	18.3673	2.0408	36.7347	14.2857
	No Idea	-	-	-	-	-	-

4.6 Discussion

In this section, we discuss our evaluation result and state of the art. In the literature the ratio of the action recognition rate for walk, run, kick, jump and punch are respectively 87%, 22%, 12%, 5% and 0% from the user study (Yumer & Mitra, 2016). The percentage values of action recognition representing in the each cell of confusion (Table 4.6). We achieve the good recognition rate for walking, running, jumping, kicking and lying. There

Table 4.6: Action recognition user-study confusion matrices. In each cell: action recognition percentage values are given.

Corrected Action \ User Selected Action	Walk	Stand	Run	Jump	Punch	Kick	Sit	Throw	Hit	Grab	Push	Drink	Dress	Eat	Lying	Talk	Fall	Cut	NoIdea
Walk	97.78	-	-	-	-	-	-	-	-	-	0.74	-	-	-	-	-	-	-	1.48
Stand	-	87.76	-	-	-	-	-	-	-	-	-	-	-	-		4.08	-	-	8.16
Run	12	-	87.20	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0.80
Jump	-	3.75	-	87.80	-	-	-	1.22		1.22	-	-	-	-	-	-	-	-	6.01
Punch	-	-	-	-	68.13	-	-	6.59	14.30		6.59		-	-	-	-	-	-	4.39
Kick	-	1.04	1.04	4.17	-	89.58	-	-	-	-	-	-	-	-	-	-	3.13	-	1.04
Sit	-	-	-	-	-	-	76.73	-	-	-	-	-	-		2.59	5.18	1.72	1.72	7.76
Throw	-	6.89	-	-	-	1.15	-	67.82	6.89	-	4.59	-	-	-	-	-	-	-	12.65
Hit	-	-	-	-	31.43	-	-	2.86	45.71	-	11.43	-	-	-	-	-	-	-	8.57
Grab	-	11.76	-	-	1.47	-	-	-	-	50	14.71	-	-	-	-	-	4.41		17.65
Push	-	1.79			8.93	-	-	-	-	28.57	39.29	-	1.79	-	-	5.36	-	-	14.29
Drink	-	3.45	-	-	-	-	-	-	-	13.79	-	60.34	-	12.07	-	1.72	-	-	8.62
Dress	-	-	-	-	-	-	-	4.54		13.64	13.64	-	45.45	-	-	-	-	-	22.73
Eat	-	-	-	-	-	-	-	-	-	-	-	30		50	-	-	-		20
Lying	-	-	-	-	-	-	-	-	-	-	-	-	-	-	83	-	-	-	17
Talk	26		-	-	-		34	-	-	-	-	-	-	-		40	-	-	-
Fall	-	-	-	48.15	-	-	-	-	-	-	-	-	-	-	-	-	51.85	-	
Cut	-	30	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	40	30
Noidea	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

is some interesting outcome of our user study is the fact that hitting, throwing and pushing are being significantly confused with hitting, throwing and pushing is presented in (Table 4.6). In this table, another action grabbing is confused with standing and pushing. Besides, eating and drinking are confused with eating drinking. Talk is confused with sitting and walking. In addition, falling is confused with jumping.

The state art of style transfer of human synthesis (Yumer & Mitra, 2016), provided style user study recognition confusion matrices, and they achieved 94% neutral and neutral is confused with proud style. Besides, Yumer and Mitra (2016) obtained 90% angry, and angry is confused with proud. Fear is confused with neutral, and sad is shown in our user study confusion matrices (Table 4.5).

4.7 Extension of the human action video signal

In this section, a research has been done to extend the length of an input video signal to any given duration. To achieve that, we need to find the period of the given human action signal and then repeat the period to generate the output video of a desired length. First, the periodic joint is selected for a given human action to calculate the period of that joint. The selection is done by calculating the distance between the highest peak and second highest peak of the power spectrum. The joint for which this distance is the maximum is considered as the periodic joint for the given human action. For example, in current dataset, right hand joint is chosen as the most periodic joint for walking and running actions. The most periodic joints for a walking action displays in Table 4.7.

In order to calculate the period of the most periodic joint, *findpeaks* method is used to locate the local peak points in the input signal. Then the highest peak location is identified, which is considered as the starting point of the periodicity. The ending point will be the last peak location provided by the *findpeaks* method. As the datasets human action signal is a composite signal, this approach provides a better estimation of the cycle value for the periodic joint. This approach also generates a smoother transition of the frames in the extended length of the video output.

The other two approaches that are used to find the periodicity of the input signal are Auto-correlation based approach (*xcorr* method in Matlab) and Fast Fourier Transform based approach (*fft* method in Matlab). These two approaches provide some estimation of the period for the given signal. But these methods work only in the cases where the periodicity starts from the beginning of the input signal. As the datasets human actions are composite signals, these methods are not a good candidate to calculate the overall period of the video signal.

Table 4.7: List of skeleton joints name which contains periodic motion

Action	**Skeleton joint**
Walking	Lefthandindex1, Lefthandindex2, Leftthumb1, Leftthumb2, Lefthand, Leftfingerbase, Leftarm,Leftupleg
Running	Leftforearm, Righthandindex2, Righthandindex1, Righthand, Rightfingerbase, Lefthand, Leftforearm, Lefthandindex1, Lefthandindex2, Leftarm, Leftfingerbase.
Punch	Leftarm, leftforearm, leftfoot,Leftfootbase1, leftfootbase2, leftupleg, leftleg,lefthand, lefthandfingerbase1, lefthandfingerbase2, lefthandindex1, lefthandindex2.
Kicking	Rightforearm, Righthandindex1, Righthandindex2, Rightthumb, Righthipjoint, Rightleg, Rightfoot, Rightfootbase, Righthipjoint
Throwing	Leftffoot, Spine, Hips, Righthipjoint, Leftforearm, Rightfoot, Leftforearm
Grabbing	Righthandthumb, Righthand, Righthandindex1, Rightforearm, Lefthandhumb, Rightleg, Lefthand

Chapter 5

Conclusion and Future Work

Realistic human action synthesis is very complex. The current state state of the art still requires an exhaustive database including all possible actions and transitions to perform a realistic style transfer. This thesis presents, synthesizing various actions with the different set of emotions by applying style or emotion transfer algorithm. This work has the advantage of psychological research on emotion besides the enhancing database and animation industry. CMU motion capture dataset has been used in this research to justify the proof-of-concept. The proposed framework can easily be extended to include other actions and emotions. In future, we would like to include more actions and emotions. Besides other emotions, we would like to investigate the ways of generating actions with different personalities, ages and genders. This research poses a good platform to use this concept in the gaming industry where actors emotion and action can easily be synthesized to design a more realistic experience for end users. In spectral domain, imposing direct spatial constraints can be the future direction of research.

References

Amaya, K., Bruderlin, A., & Calvert, T. (1996). Emotion from motion. In *Graphics interface* (Vol. 96, pp. 222–229).

Biovision bvh. (n.d.). Retrieved from `https://research.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/BVH.html` ([Online; accessed 7th-February-2017])

Brand, M., & Hertzmann, A. (2000). Style machines. In *Proceedings of the 27th annual conference on computer graphics and interactive techniques* (pp. 183–192).

Du, C. H. (2014). *A generative statistical model for human motion synthesis* (Unpublished doctoral dissertation). Saarland University.

Fang, A. C., & Pollard, N. S. (2003). Efficient synthesis of physically valid human motion. In *Acm transactions on graphics (tog)* (Vol. 22, pp. 417–426).

Freeman, W. T., Tenenbaum, J. B., & Pasztor, E. C. (2003). Learning style translation for the lines of a drawing. *ACM Transactions on Graphics (TOG)*, *22*(1), 33–46.

Hahne, B. (2010). *The motionbuilder-friendly bvh conversion release of cmu's motion capture database.* Retrieved from `https://sites.google.com/a/cgspeed.com/cgspeed/motion-capture/cmu-bvh-conversion` ([Online; accessed 7th-February-2017])

Holden, D., Saito, J., & Komura, T. (2016). A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics (TOG)*, *35*(4), 138.

Hsu, E., Pulli, K., & Popović, J. (2005). Style translation for human motion. In *Acm transactions on graphics (tog)* (Vol. 24, pp. 1082–1089).

Kovar, L., & Gleicher, M. (2003). Flexible automatic motion blending with registration curves. In *Proceedings of the 2003 acm siggraph/eurographics symposium on computer animation* (pp. 214–224).

Kovar, L., Gleicher, M., & Pighin, F. (2002). Motion graphs. In *Acm transactions on graphics (tog)* (Vol. 21, pp. 473–482).

Lhommet, M., & Marsella, S. C. (2014). Expressing emotion through posture. *The Oxford Handbook of Affective Computing*, 273.

Lonkar, S. (2017). *Types of motion capture.* Retrieved from `https://sagarlonkar.com/about-2/motion-capture/types-of-motion-capture/` ([Online; accessed 26-Feb-2017])

Max-Planck Institute for Biological Cybernetics in Tuebingen, G. (2014). *Emotional Body Motion Database.* `http://ebmdb.tuebingen.mpg.de/`. ([Online; accessed 4-May-2017])

Meredith, M., Maddock, S., et al. (2001). Motion capture file formats explained. *Department of Computer Science, University of Sheffield*, *211*, 241–244.

Min, J., Liu, H., & Chai, J. (2010). Synthesis and editing of personalized stylistic human motion. In *Proceedings of the 2010 acm siggraph symposium on interactive 3d graphics and games* (pp. 39–46).

Mizuguchi, M., Buchanan, J., & Calvert, T. (2001). Data driven motion transitions for interactive games.

Oppenheim, A. V., & SCHAFER. (2009). *Discrete-time signal processing* (Vol. 3). Prentice Hall.

Optical Motion Capture Systems. (2017). `http://metamotion.com/motion-capture/optical-motion-capture-1.htm`. ([Online; accessed 26-Feb-2017])

Starbuck, R., Seo, J., Han, S., & Lee, S. (2014). A stereo vision-based approach to marker-less motion capture for on-site kinematic modeling of construction worker tasks. In *Computing in civil and building engineering (2014)* (pp. 1094–1101).

Team, M. (n.d.). Makehuman. Retrieved from `http://makehuman.org`

Vosk, B. N., Forehand, R., & Figueroa, R. (1983). Perception of emotions by accepted and rejected children. *Journal of Psychopathology and Behavioral Assessment*, *5*(2), 151–160.

Wang, X., Chen, Q., & Wang, W. (2014). 3d human motion editing and synthesis: a survey. *Computational and mathematical methods in medicine*, *2014*.

Wooten, W. L., & Hodgins, J. K. (2000). Simulating leaping, tumbling, landing and balancing humans. In *Robotics and automation, 2000. proceedings. icra'00. ieee international conference on* (Vol. 1, pp. 656–662).

Xia, S., Wang, C., Chai, J., & Hodgins, J. (2015). Realtime style transfer for unlabeled heterogeneous human motion. *ACM Transactions on Graphics (TOG)*, *34*(4), 119.

Yumer, M. E., & Mitra, N. J. (2016). Spectral style transfer for human motion between independent actions. *ACM Transactions on Graphics (TOG)*, *35*(4), 137.