Synthesizing Motion with Relative Emotion Strength

Edmond S. L. Ho
Northumbria University, UK
e.ho@northumbria.ac.uk

Hubert P. H. Shum
Northumbria University, UK
hubert.shum@northumbria.ac.uk

He Wang
University of Leeds, UK
H.E.Wang@leeds.ac.uk

Li Yi
Yilifilm, China
yilistudio@126.com

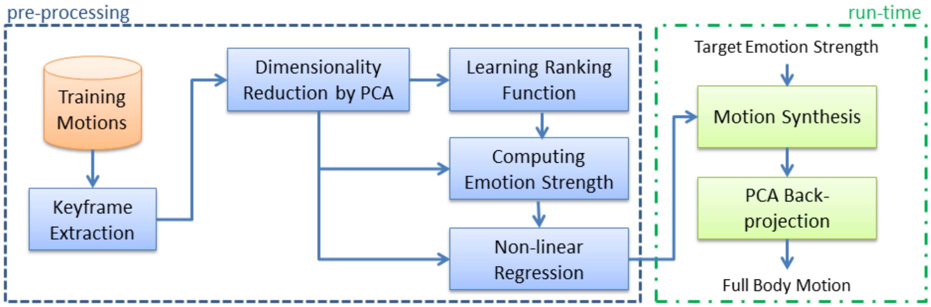


Figure 1: Overview of the proposed methodology.

ABSTRACT

With the advancement in motion sensing technology, acquiring high-quality human motions for creating realistic character animation is much easier than before. Since motion data itself is not the main obstacle anymore, more and more effort goes into enhancing the realism of character animation, such as motion styles and control. In this paper, we explore a less studied area: the emotion of motions. Unlike previous work which encode emotions into discrete motion style descriptors, we propose a continuous control indicator called *motion strength*, by controlling which a data-driven approach is presented to synthesize motions with fine control over emotions. Rather than interpolating motion features to synthesize new motion as in existing work, our method explicitly learns a model mapping low-level motion features to the emotion strength. Since the motion synthesis model is learned in the training stage, the computation time required for synthesizing motions at run-time is very low. As a result, our method can be applied to interactive applications such as computer games and virtual reality applications, as well as offline applications such as animation and movie production.

CCS CONCEPTS

• **Computing methodologies** → **Animation**; *Motion capture*;

KEYWORDS

motion capture, data-driven, motion synthesis, emotion motion, relative attribute

ACM Reference format:
Edmond S. L. Ho, Hubert P. H. Shum, He Wang, and Li Yi. 2017. Synthesizing Motion with Relative Emotion Strength. In *Proceedings of ACM SIGGRAPH ASIA Workshop: Data-Driven Animation Techniques (D2AT), Bangkok, Thailand, Nov 2017 (D2AT'17),* 8 pages.
https://doi.org/10.475/123_4

1 INTRODUCTION

Synthesizing realistic human motion from existing motion data, either acquired by motion capture (MOCAP) devices or hand-crafted, has been an active research area in the past two decades. In graphics applications such as movies, animation, interactive computer games and virtual reality (VR), realism of human motion plays an important role in enhancing the user's experience. In the past, motion naturalness and the ease of control have been intensively studied. While they make the character animation tasks easier, we argue that naturalness and control alone are not enough to generate realistic and diversified motions. Emotion is another indispensable piece in motion realism. In literature, there is an increasing focus on analyzing how motion emotion are perceived by users [Aristidou and Chrysanthou 2013; Ennis et al. 2013; McDonnell et al. 2008; Normoyle et al. 2013]. While it is possible to identify emotion automatically from motion, synthesizing motion with controllable emotion is much more challenging due to the high-dimensionality in the control parameters as well as the complex relationship between such control parameters and the emotion expressions.

The most relevant stream of work is style transfer from example motions to a new one [Xia et al. 2015; Yumer and Mitra 2016].

D2AT'17, Nov 2017, Bangkok, Thailand

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

However, the existing work solves this problem by interpolating sample motions. We argue that a drawback of such an approach is *interpolating the motion but not necessarily interpolating the emotion.* In other words, existing methods mainly learn low-level motions features in the way that the control of interpolation only happens at the motion level, not at the emotion level. This is because the learning methods ignore the relations between low-level motion features and high-level emotions. In this work, we tackle this problem by learning the relationship between low-level motion features (such as joint positions, velocities) and high-level emotion attributes with different levels of strengths.

Learning the relationship between low-level motion features and mid-/high-level semantic representation has been explored in the computer vision community. *Attributes* [Ferrari and Zisserman 2008] have been widely used in visual recognition tasks. In particular, relative attributes [Parikh and Grauman 2011] that indicate the relative strength of an attribute between two samples can naturally be used to describe the differences in high-level semantics such as styles and emotions among different motions. The relationship between motion features and the attribute can be learned from the training data to facilitate motion synthesis.

In this paper, we propose a new data-driven method to synthesize motion with controlled emotion expressions. The goal is to synthesize motions with emotions intuitively (e.g. directly specifying the level of *happiness* (i.e a scalar scale) of the motion to be edited). Our method treats the emotion of each motion as a high-level representation (i.e. attribute) and learns the relative strength of the emotion by pairwise relative attribute. A scoring function that evaluates the attribute strength will be learned and used to guide the motion synthesis process. To our knowledge, our method is the first attempt to learn the relationship between low-level motion features and human nameable attributes for motion synthesis instead of interpolating motion directly. Since the learning process is done in the pre-processing stage, the run time computational cost is very low. As a result, our method can be applied to interactive applications such as computer games and VR applications, as well as offline applications such as animation and movie production.

1.1 Contributions

The main contributions of this work can be summarized as follows:

- The first method to learn the relationship between low-level motion features and the strength of emotion expressions for human motion synthesis
- A real-time motion synthesis framework that is controlled solely by the strength of emotion expressions

2 RELATED WORK

In this section, we will first review the related work in editing and synthesizing human motion with styles, which creates new motions based on high-level properties. Next, we review the work in learning the relationship between high-level semantic representations and low-level features from the input data.

2.1 Style-based motion synthesis and editing

An early work by [Brand and Hertzmann 2000] proposed to synthesize styled motion sequences by interpolating and extrapolating the styles from the training motion data using a learned statistical model called *style machine.* [Urtasun et al. 2004] proposed an approach to project input motion to lower dimensional space by Principal Component Analysis (PCA) and compare the difference with other example motions in the database. The differences are then to be used as the style of the input motion and transferred to other motions. [Hsu et al. 2005] proposed to learn the difference in the styles in motion pairs (e.g. a neutral and a styled motion as input and output styles). The relationship between the input and output motions in training can then be described by a linear time-invariant (LTI) model. Using such a model, the learned style can be transferred to a new input motion. [Ikemoto et al. 2009] proposed a method for editing motion using Gaussian process models of dynamics and kinematics. Such an approach can be used for motion style transfer. [Xia et al. 2015] proposed a real-time approach for style transfer for human motion. Their method represents the style difference between the input and output motions by a time-varying mixture of autoregressive models. Their method learns such models automatically from unlabeled heterogeneous motion data.

While many style editing and transferring approaches have been proposed, most of the work focuses on finding the differences between motions with different styles and creating new motion by interpolation. We argue that such an approach does not necessarily interpolate the underlying emotion of the motion. In this work, we learn the relationship between motion features and emotion strength such that the motion synthesis process can be directly controlled using the target emotion strength.

2.2 Attribute-based representations

Visual attributes are human-nameable semantic concepts of things in the world and are popular in the computer vision community for image recognition [Ferrari and Zisserman 2008]. [Parikh and Grauman 2011] further proposed the concept of the relative attribute to capture the relative attribute strength between data. The aforementioned approaches mainly focused on handling visual recognition tasks (e.g. image classification). A recent work showed that images (e.g. 2D outdoor scenes) can be edited by applying the relative attribute concepts [Laffont et al. 2014]. Another recent work learns the mapping between high-level descriptors (e.g. softness, silkiness) and simulation parameters for cloth simulation [Sigal et al. 2015]. A perceptual control space is then created and the cloth simulation can be controlled by adjusting the degrees in selected high-level descriptors. These recent studies motivated us to learn such a relative attribute strength for motion synthesis.

To sum up, most of the existing work focuses on learning the attribute strength from input data for recognition tasks. While some recent work applied the concept of attribute-based editing in the computer graphics community, our work is the first to make use of the attribute strength for human motion editing.

3 OVERVIEW

The overview of our proposed methodology is illustrated in Figure 1. There are two stages in the proposed framework: pre-processing and run-time stages. The pre-processing stage learns models for analyzing and synthesizing motions with different emotion strength. Firstly, we collect motions with emotion labels as the input of our framework. Next, the motions are represented compactly by keyframing and dimensionality reduction. Then, the relative emotion strength of the training motion data is computed by a ranking function. Finally, the relative emotion strength and compact motion representation will be used for training a motion synthesis model.

During run-time, the target emotion strength (i.e. a scalar value) will be used as input to control how the motion will be synthesized using the model trained in the pre-processing stage. Full body motion can be obtained by back-projecting the synthesized compact motion representation to full body motion space.

4 LEARNING ATTRIBUTE STRENGTH

In this section, we present the core methodology of the new data-driven motion synthesis approach that learns the relationship between low-level motion features and the emotion strength. Firstly, the motion data being used in our approach for learning will be explained in Section 4.1. Next, we propose to use a compact motion representation that removes the temporal redundancy of the motion in Section 4.2. Thirdly, the dimensionality of the frame-based (i.e. pose) representation (Section 4.2.3) is further reduced to facilitate the learning process in a later stage. Finally, the learning process of the ranking function that computes the emotion strength will be explained in Section 4.3.

4.1 Emotion Motions

In order to analyze and learn the relationship between low-level motion features and the strength of emotion expressions, a significant amount of human motion data has to be collected. In this work, the MOCAP data from the Body Movement Library [Ma et al. 2006] was used. This dataset contains 4080 human motion sequences captured using a commercial optical motion capture system. The motions were captured from 30 (15 male and 15 female, ranging from 17 to 29 years old and with a mean age of 22 years old) *non-professional participants* in order to capture natural emotion expressions rather than capturing staged/exaggerated motions from professional performers. There are three motion types: *knocking, lifting,* and *throwing*.

Each motion contains a sequence of poses captured from a single subject and represented by the joint positions in Cartesian coordinates in each frame. From the raw motion data, we extracted 33 joint positions in each frame. The duration of each motion sequence ranges from 65 to 457 frames at 60 Hz. The dataset was designed for recognizing identity, gender and emotion from the motion data. As a result, each subject performed each motion type with 4 different emotion expressions: *Neutral, Angry, Happy,* and *Sad* .

In this work, we focus on using two emotion models, namely *Happiness* and *Anger*, because a continuous emotion parameterization can be achieved. Specifically, for the *Happiness* model, we define *Happy* and *Sad* to be opposite to each other and *Neutral* to be at the middle. For the *Anger* model, we define *Anger* and *Neutral* as two extremes. As a result, we can simplify the parameterization as a single scalar value to control the emotion level continuously in *Happiness* (*Happy* ↔ *Neutral* ↔ *Sad*) and *Anger* (*Angry* ↔ *Neutral*). This allows us to control the emotion strength continuously.

4.2 Motion Representation

Having presented the details of the motion data for training, we now explain the compact motion representation used in our proposed framework. Each motion M contains a sequence of poses p, i.e. $M = \{p_1, ..., p_n\}$, where n is the total number of frames (or poses) of M. Each pose p_i is represented by the 3D joint positions in Cartesian coordinates. As there are 33 joints extracted from the raw motion data, each pose is represented by a 99-dimensional vector. In order to facilitate the learning process, we normalize all data by removing the translation and the vertical rotation (i.e. y-axis) of the root joint in the first frame as in other data-driven approaches (e.g. [Ho et al. 2016]).

4.2.1 *Keyframe Extraction.*

Human motions are naturally temporally redundant and removing such redundancy can facilitate the the learning process in later stages. In addition, the motion sequences we collected have different durations. As a result, we cannot carry out machine learning from the training motion data directly. To tackle the aforementioned problems, keyframes of each motion are extracted by Curve Simplification algorithm [Lim and Thalmann 2001]. Given a motion M, a set of q keyframes $K = \{k_1, ...k_q\}$ will be extracted, which minimizes the reconstruction error when interpolating the in-between motion using spline interpolation. In this study, we tested the reconstruction error with 10-15 keyframes and empirically found that using 13 keyframes can balance the trade-off between reconstruction error and compactness of the motion representation.

4.2.2 *Joint Velocity.*

We observed that some of the subjects expressed different emotions by using different speed and rhythm when performing the motion. Using the *knocking door* motion as an example. The subjects tended to move faster when they were *happy*, and moved slower when they were *sad*. For this reason, the velocity of the joints between adjacent keyframes is computed:

$$v_{i+1} = \frac{k_{i+1} - k_i}{\Delta t} \tag{1}$$

where Δt is the duration between the two adjacent keyframes. Here, we compute the joint velocity from adjacent keyframes instead of adjacent frames in the original motion because we want to reconstruct the full motion into different durations by editing the velocity. Specifically, given the adjacent keyframes and the joint velocity, the duration between two adjacent keyframes can be approximated by:

$$\Delta t = \frac{k_{i+1} - k_i}{v_{i+1}} \tag{2}$$

By this, the speed of the motion can be adjusted easily using this compact representation. Finally, each keyframe contains the pose features k_i and velocity features v_i and results in a 99 + 99 = 198-dimensional feature vector.

4.2.3 *Dimensionality Reduction.*

To further reduce the dimensionality of the pre-frame (i.e. pose) feature to facilitate the learning process (will be explained in Section 4.3), the pose features and velocity features are concatenated and projected to a low-dimensional space. While many dimensionality reduction approaches are available, Principal Component Analysis (PCA) has been used for analyzing the style difference for motion synthesis [Urtasun et al. 2004].

This leads us to use such a simple, low-computational cost and widely used dimensionality reduction techniques in the proposed method.

One of the issues is to select an appropriate dimensionality in the latent space such that the essential information in the original motion is retained. We empirically calculate the reconstruction error (i.e. back-projecting the latent representation and compare the result with the keyframe features) using different numbers of dimension in the latent space. We found that projecting the keyframe features from 198-d to 40-d achieved low reconstruction error and we use this setting in all experiments.

4.3 Learning the Ranking Function

In this section, we will explain how to learn the relationship between the latent representation and the strength of the attribute from two sets of inputs: i) the compact motion features explained in Section 4.2.3 and ii) the emotion label associated with each motion. One simple way to learn such a function is to train a regression function on the ground truth emotion strength and the corresponding motion features. However, the ground truth attribute strength may not be available in the database. In the Body Movement Library database we used, only a single label (i.e. the emotion) is associated with each motion. The relative emotion strength between motions with the same class label is not available.

Inspired by [Parikh and Grauman 2011], a ranking function can be learned from a small set of pairwise training samples with relative ranking on an attribute (i.e. emotion in our framework). Such a ranking function can then be used for computing the attribute strength of unseen data. The learning process can be formulated as a max-margin optimization problem. Specifically, we learn a ranking function w to weight each input feature and return the weighted sum as the attribute score that indicates the attribute strength. When solving for the ranking function, a set of *relative constraints* have to be satisfied. Using the notation in [Parikh and Grauman 2011]:

$$\forall(i,j) \in O : wx_i > wx_j \tag{3}$$

$$\forall(i,j) \in S : wx_i = wx_j \tag{4}$$

where O and S are sets that contain *ordered* and *similar* paired samples, and x_i and x_j are the feature vectors of the i-th and j-th samples (motions in our approach).

More specifically, the ordered set O contains motions with difference in ground truth attribute strength in each pair as in Eq. 3. In the dataset we used, we setup the ordered pairwise relative constraints according to the emotion labels of the training motions: *Happy* > *Neutral* > *Sad* and *Angry* > *Neutral* when training the *Happiness* and *Anger* ranking functions, respectively. On the other hand, the similar set S contains motions with similar ground truth attribute strength in each pair as in Eq. 4. The similar set contains the pairwise motions with the same emotion label in our proposed framework. The attribute strength of each sample can be computed by multiplying the ranking function w with the feature vector (e.g. x_i or x_j) and our task is to learn w by:

$$\begin{aligned} \min_{w} \quad & \frac{1}{2}\|w\|_2^2 + C\Big(\sum \xi_{ij}^2 + \sum \gamma_{ij}^2\Big) \\ \text{s.t.} \quad & w(x_i, x_j) \geq 1 - \xi_{ij}; \forall(i,j) \in O \\ & |w(x_i, x_j)| \leq \gamma_{ij}; \forall(i,j) \in S \\ & \xi_{ij} \geq 0; \gamma_{ij} \geq 0, \end{aligned} \tag{5}$$

where C is the trade-off parameter to control the softness of the pairwise relative constraints to be satisfied, and ξ_{ij} and γ_{ij} are slack variables. This primal problem can be solved efficiently by Newton's method [Chapelle 2007].

5 MOTION SYNTHESIS BY ATTRIBUTE STRENGTH

Having learned the ranking function as explained in Section 4.3, the emotion strength (a scalar) of each training motion can be computed. Since our ultimate goal is to synthesize new motion by specifying the emotion strength, here we propose to learn a regression model on the emotion strength and the corresponding motion features from the training data. Specifically, we learn a regression function $f(s)$ that takes the target emotion strength s as input and returns the dimensionality reduced motion feature x:

$$x = f(s) \tag{6}$$

In the implementation of the proposed framework, we train the regression function using the Neural Network regression model in MATLAB [MATLAB 2015]. The implementation details will be explained in Section 6.1.

6 EXPERIMENTAL RESULTS

In this section, we present the results obtained using the proposed method. Firstly, we visualize the motions projected to a low-dimensional space when training our motion synthesis models in Section 6.2. Next, we evaluate the performance of the learned ranking functions that contain the relationship between low-level motion features and emotion strength numerically in Section 6.3. Thirdly, we present the computational cost for the training process in Section 6.4. Finally, new motions are synthesized using the learned motion synthesis models in Section 6.5.

6.1 Implementation Details

We used the Toolbox for Dimensionality Reduction library [van der Maaten et al. 2009] to reduce the feature dimension from 198-d to 40-d. For the relative attribute ranking, we used the MATLAB implementation provided by the authors of [Parikh and Grauman 2011]. The proposed framework was implemented in Matlab R2015a [MATLAB 2015] and all the experiments were conducted on a 64-bit machine with Intel Xeon 2.4GHz (E5-2620) and 64GB memory. The experiments ran on a single thread without any performance boost.

6.2 Visualization of the Low-dimensional Space

In order to learn the ranking function efficiently and effectively, the motion features are projected into latent space with a much lower dimensionality. In the experiments, each pose (frame) is represented by a 40-d in the latent space (198-d in the original motion). The dimensionality in the latent representation is selected empirically to balance the trade-off between training time and reconstruction error rate. We use 10 as the hidden layer size in the Neural-Network regression for training the motion synthesis model. The latent representation of the motions with different emotions are illustrated in Figure 2 and 3 (left columns) (only the first 3 principal components are displayed).

However, due to the style differences and variations between the subjects, the motions (represented as curves) with different emotion labels are mixed together. This shows that directly interpolating motions in the low dimensional space may result in significant change in the emotion strength as the motions (i.e. trajectories) with different emotions (i.e. colors) are tangled.

To facilitate motion synthesis by controlling the emotion strength, our proposed framework must be able to learn the important motions features that affect the emotion strength. The relationship can be learned by training a ranking function as explained in Section 4.3. The ranking function will be used for calculating the motion strength (a scalar value) of each motion. Then, the motions are ranked by the motion strength as shown in Figure 2 and 3 (right columns). The results indicate that the learned ranking function can effectively evaluate the emotion strength of each motion and is able to separate motions from different emotion strength. This facilitates the motion synthesis process as the ranking function contains the weights of each motion feature that reflects how much each feature contributes to the change in the emotion strength.

6.3 Evaluating the Learned Ranking Function

In addition to visualizing the effectiveness of the learned ranking functions on evaluating the emotion strength, we further evaluate the performance of the ranking function numerically. Specifically, we split the collected motions into 2 sets with equal numbers of motions - one set for training and the other set for testing such that the testing motions are 'unseen' data to the ranking function. Next, we use the labels of the training motions to setup the relative constraints as explained in Section 4.3. Then, the learned ranking function is used for computing the emotion strength of the testing motions. Finally, we compare the computed relative emotions strength to the labels of the testing motions and obtain the accuracy in ranking the testing motions. The results are shown in Table 1 and 2. They indicate that high ranking accuracy can be achieved in which the learned ranking function can be generalized to unseen data. This highlights the robustness of the ranking function. Moreover, the results also show that using PCA to reduce the dimensionality of the frame-based features does not have a significant impact on the ranking accuracy while improving the efficiency of the training process. We also vary the number of relative constraints used to train the ranking function and the results show that there are no significant differences in the ranking accuracy.

6.4 Computational Cost for the Training Tasks

In this section, we present the computational costs for the model training tasks. Table 3 shows the computation time required for training the ranking function explained in Section 4.3. The results show a significant reduction in training time when compared with the full feature setup (i.e. no PCA). This further highlights the performance gain using the proposed dimensionality reduction approach.

motion type	frame feature	% of training pairs				
		10%	30%	50%	70%	90%
Knocking	PCA 40-d	71.3%	70.1%	69.5%	69.2%	68.7%
	PCA 80-d	71.9%	71.2%	70.8%	70.5%	70.4%
	no PCA	72.5%	72.0%	71.7%	71.5%	71.3%
Lifting	PCA 40-d	72.7%	73.1%	73.1%	73.1%	73.3%
	PCA 80-d	72.1%	72.9%	73.1%	73.1%	72.9%
	no PCA	72.3%	72.8%	72.8%	72.7%	72.8%
Throwing	PCA 40-d	75.2%	75.6%	76.0%	76.0%	76.1%
	PCA 80-d	75.3%	75.6%	75.9%	75.9%	76.1%
	no PCA	75.3%	75.7%	75.9%	75.9%	76.0%

Table 1: Ranking accuracy of the trained anger ranking function on unseen motion data.

motion type	frame feature	% of training pairs				
		10%	30%	50%	70%	90%
Knocking	PCA 40-d	78.3%	76.6%	75.9%	75.2%	75.0%
	PCA 80-d	80.6%	79.3%	78.9%	78.5%	78.4%
	no PCA	81.1%	80.1%	80.0%	79.6%	79.7%
Lifting	PCA 40-d	79.9%	79.1%	78.6%	78.1%	77.7%
	PCA 80-d	80.9%	81.0%	81.0%	80.9%	80.8%
	no PCA	81.4%	81.3%	81.3%	81.3%	81.1%
Throwing	PCA 40-d	76.5%	76.0%	75.3%	74.9%	74.7%
	PCA 80-d	76.8%	76.7%	76.2%	76.0%	75.8%
	no PCA	76.8%	76.8%	76.3%	76.0%	75.8%

Table 2: Ranking accuracy of the trained happiness ranking function on unseen motion data.

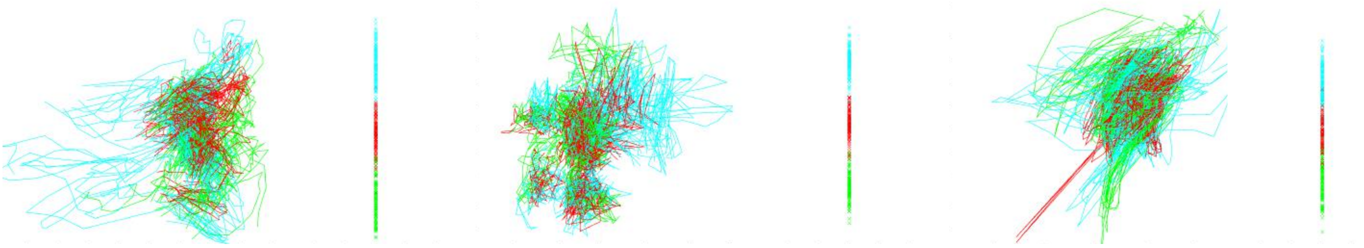
motion type	emotion model	computation time (s)		
		PCA 40-d	PCA 80-d	no PCA
Knocking	Anger	3.09	15.09	152.41
	Happiness	5.13	15.27	106.17
Lifting	Anger	2.33	14.77	116.75
	Happiness	5.43	17.40	64.48
Throwing	Anger	2.70	15.02	108.00
	Happiness	5.05	17.91	70.08

Table 3: Computational cost (in seconds) for training the ranking function using different motion features.

Table 4 shows the computation time required for training the non-linear regression function explained in Section 5. The training time varies from 10 to 15.5 minutes. Nevertheless, this training process is performed in the pre-processing stage and thus the training time is acceptable. A perform gain is expected when a parallel implementation of non-linear regression is used.

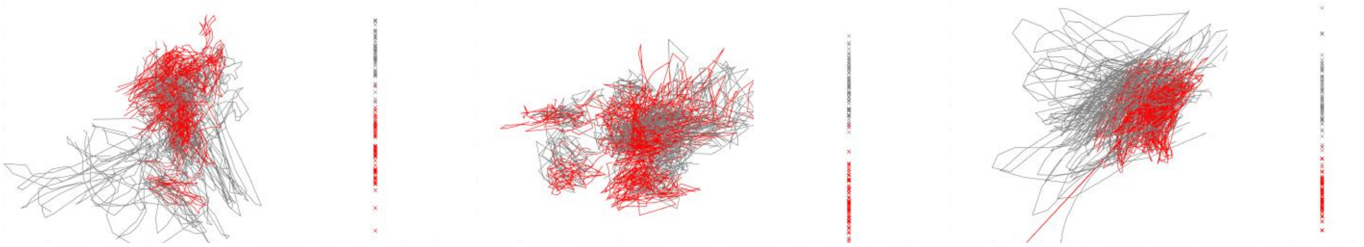
6.5 Synthesizing Motions with Different Emotion Strength

In this section, we show a number of motions with different emotion strength synthesized by our proposed framework. As explained



(a) Throwing Motion (b) Knocking Motion (c) Lifting Motion

Figure 2: Visualizing the closeness of the training motions in (left column in each motion type) latent space (projected by PCA) and (right column) sequential order based on the emotion strength computed using the learned happiness ranking function on different motion types. Motions are colored according to the ground truth emotion labels: Happy (cyan), Neutral (red) and Sad (green).



(a) Knocking Motion (b) Lifting Motion (c) Throwing Motion

Figure 3: Visualizing the closeness of the training motions in (left column in each motion type) latent space (projected by PCA) and (right column) sequential order based on the emotion strength computed using the learned anger ranking function on different motion types. Motions are colored according to the ground truth emotion labels: Angry (grey) and Neutral (red).

motion type	emotion model	computation time (s)
		PCA 40-d
Knocking	Anger	491.01
	Happiness	601.83
Lifting	Anger	915.60
	Happiness	631.12
Throwing	Anger	889.16
	Happiness	647.27

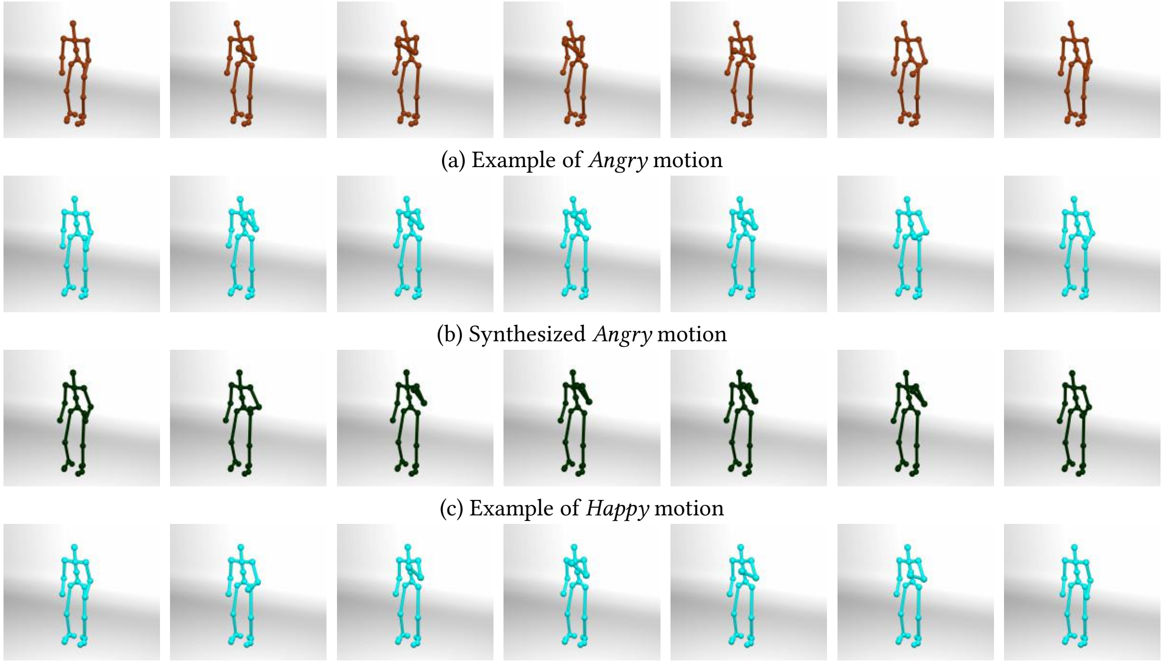
Table 4: Computational cost (in seconds) for training the non-linear regression function using different motion features.

in Section 5, a regression function is trained using the emotion strength and features obtained from the training data. At run-time, new motions can be created by specifying target emotion strength. We compared the synthesized motions with the training data at different emotion strength. Screen shots of training data and synthesized motions are shown in Figures 4, 5 and 6. The results show that our proposed method can synthesize motions with emotions that are comparable to the training data. More results can be found in the video demo accompanying with this paper.

7 CONCLUSION AND DISCUSSIONS

In summary, this paper presents a new data-driven approach to learn the underlying relationship between low-level motion features and high-level emotion expressions at different level of strength. Our method takes the advantages of using relative attribute [Parikh and Grauman 2011] to learn a ranking function that evaluates the attributes strength from low-level motion features. Since only weak relative constraints are used in the training process, the training data is not necessarily labeled with ground truth attribute strength as in the data set we used. Our method further makes use of the computed attribute strength for motion synthesis. Once the model is learned in the pre-processing stage, a new motion can be synthesized by specifying the target attribute strength at run-time. Our method can be applied to a wide variety of applications such as animation and movie production, as well as interactive applications such as computer games and virtual reality applications.

As the first attempt and a preliminary study on analyzing and utilizing learned attribute strength for motion synthesis, we only focused on learning a single attribute from the data. However, as shown in previous studies in recognition tasks [Chen et al. 2014], taking into account the correlations between multiple attributes can improve the attribute learning performance. In our application, such an approach could enable the synthesis of a motion with multiple types of emotion expression and this will be an interesting



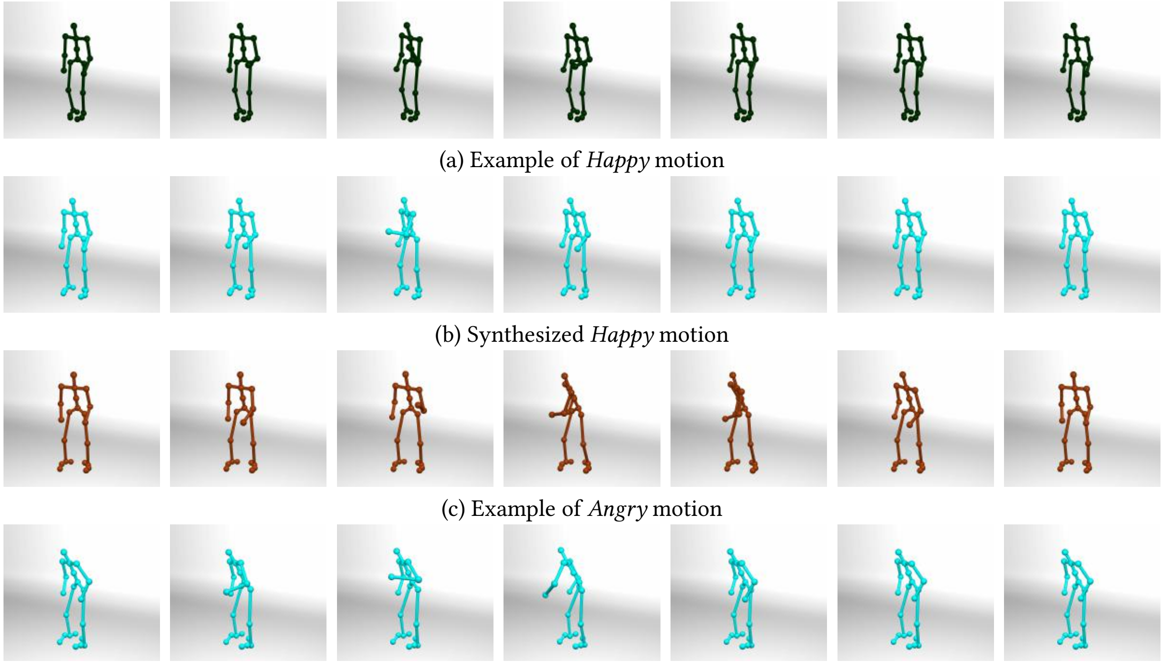
(a) Example of *Angry* motion

(b) Synthesized *Angry* motion

(c) Example of *Happy* motion

(d) Synthesized *Happy* motion

Figure 4: Knocking motions with (a) ***Angry*** **and (c)** ***Happy*** **emotion expressions in the training data and the corresponding synthesized motions (b) and (d) created by our proposed method.**



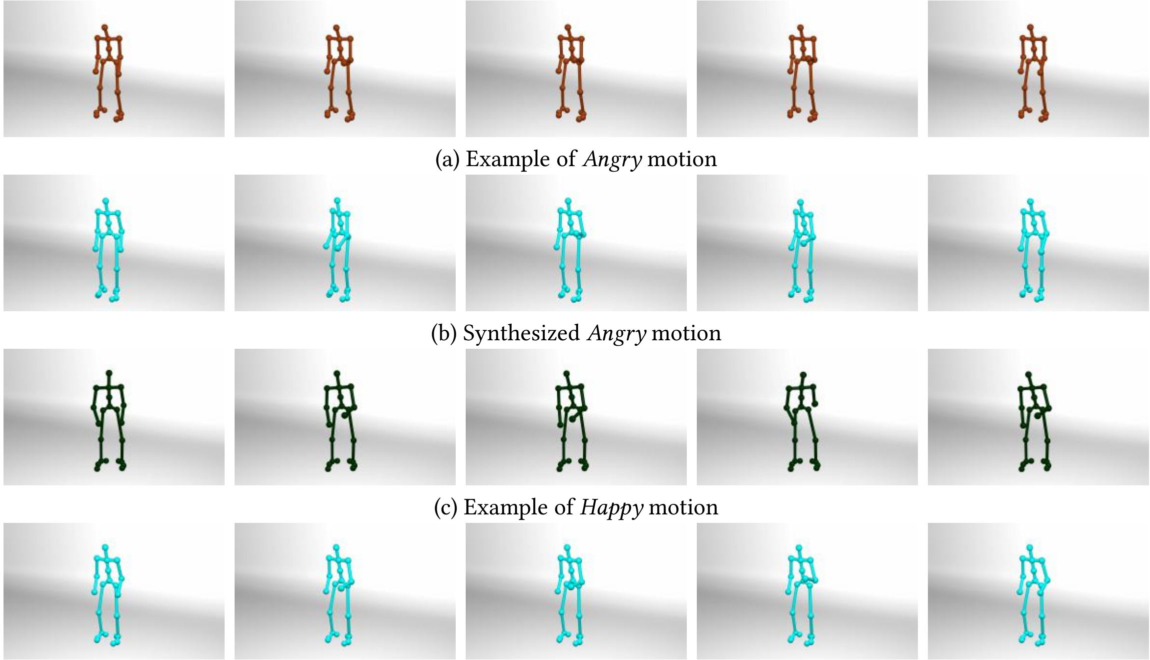
(a) Example of *Happy* motion

(b) Synthesized *Happy* motion

(c) Example of *Angry* motion

(d) Synthesized *Angry* motion

Figure 5: Throwing motions with (a) ***Angry*** **and (c)** ***Happy*** **emotion expressions in the training data and the corresponding synthesized motions (b) and (d) created by our proposed method.**



(a) Example of *Angry* motion

(b) Synthesized *Angry* motion

(c) Example of *Happy* motion

(d) Synthesized *Happy* motion

Figure 6: Lifting motions with (a) *Angry* and (c) *Happy* emotion expressions in the training data and the corresponding synthesized motions (b) and (d) created by our proposed method.

future research direction. Another future direction is conducting a user study to evaluate the emotion strength level of motions for validating the ranking function learned by our method. We are also planning to explore other possible motion features to be included in the learning process.

ACKNOWLEDGEMENTS

This work is supported in part by the Engineering and Physical Sciences Research Council (EPSRC) (Ref: EP/M002632/1) and the Royal Society (Ref: IE160609). We also thank Donald Chan and Jacky Chan for conducting the experiments.

REFERENCES

Andreas Aristidou and Yiorgos Chrysanthou. 2013. Motion Indexing of Different Emotional States Using LMA Components. In *SIGGRAPH Asia 2013 Technical Briefs (SA '13)*. ACM, New York, NY, USA, Article 21, 4 pages.

Matthew Brand and Aaron Hertzmann. 2000. Style Machines. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 183–192.

Olivier Chapelle. 2007. Training a Support Vector Machine in the Primal. *Neural Comput.* 19, 5 (May 2007), 1155–1178.

L. Chen, Q. Zhang, and B. Li. 2014. Predicting Multiple Attributes via Relative Multi-task Learning. In *2014 IEEE Conference on Computer Vision and Pattern Recognition.* 1027–1034.

Cathy Ennis, Ludovic Hoyet, Arjan Egges, and Rachel McDonnell. 2013. Emotion Capture: Emotionally Expressive Characters for Games. In *Proceedings of Motion on Games (MIG '13)*. ACM, New York, NY, USA, Article 31, 8 pages.

Vittorio Ferrari and Andrew Zisserman. 2008. Learning visual attributes. In *Advances in Neural Information Processing Systems.* 433–440.

Edmond S.L. Ho, Jacky C.P. Chan, Donald C.K. Chan, Hubert P.H. Shum, Yiu-ming Cheung, and Pong C. Yuen. 2016. Improving Posture Classification Accuracy for Depth Sensor-based Human Activity Monitoring in Smart Environments. *Computer Vision and Image Understanding* 148, C (July 2016), 97–110.

Eugene Hsu, Kari Pulli, and Jovan Popović. 2005. Style Translation for Human Motion. *ACM Trans. Graph.* 24, 3 (July 2005), 1082–1089.

Leslie Ikemoto, Okan Arikan, and David Forsyth. 2009. Generalizing Motion Edits with Gaussian Processes. *ACM Trans. Graph.* 28, 1, Article 1 (Feb. 2009), 12 pages.

Pierre-Yves Laffont, Zhile Ren, Xiaofeng Tao, Chao Qian, and James Hays. 2014. Transient Attributes for High-level Understanding and Editing of Outdoor Scenes. *ACM Trans. Graph.* 33, 4, Article 149 (July 2014), 11 pages.

Ik Soo Lim and D. Thalmann. 2001. Key-posture extraction out of human motion data. In *Engineering in Medicine and Biology Society, 2001. Proceedings of the 23rd Annual International Conference of the IEEE*, Vol. 2. 1167–1169 vol.2.

Yingliang Ma, Helena M. Paterson, and Frank E. Pollick. 2006. A motion capture library for the study of identity, gender, and emotion perception from biological motion. *Behavior Research Methods* 38, 1 (2006), 134–141.

MATLAB. 2015. *version 8.5.0 (R2015a).* The MathWorks Inc., Natick, Massachusetts.

Rachel McDonnell, Sophie Jörg, Joanna McHugh, Fiona Newell, and Carol O'Sullivan. 2008. Evaluating the Emotional Content of Human Motions on Real and Virtual Characters. In *Proceedings of the 5th Symposium on Applied Perception in Graphics and Visualization (APGV '08)*. ACM, New York, NY, USA, 67–74.

Aline Normoyle, Fannie Liu, Mubbasir Kapadia, Norman I. Badler, and Sophie Jörg. 2013. The Effect of Posture and Dynamics on the Perception of Emotion. In *Proceedings of the ACM Symposium on Applied Perception (SAP '13)*. ACM, New York, NY, USA, 91–98.

D. Parikh and K. Grauman. 2011. Relative attributes. In *2011 International Conference on Computer Vision.* 503–510.

Leonid Sigal, Moshe Mahler, Spencer Diaz, Kyna McIntosh, Elizabeth Carter, Timothy Richards, and Jessica Hodgins. 2015. A Perceptual Control Space for Garment Simulation. *ACM Trans. Graph.* 34, 4, Article 117 (July 2015), 10 pages.

Raquel Urtasun, Pascal Glardon, Ronan Boulic, Daniel Thalmann, and Pascal Fua. 2004. Style-Based Motion SynthesisâĂă. *Computer Graphics Forum* 23, 4 (2004), 799–812.

L.J.P. van der Maaten, E. O. Postma, and H. J. van den Herik. 2009. Dimensionality Reduction: A Comparative Review. *J Mach Learn Res* 10 (2009), 66–71.

Shihong Xia, Congyi Wang, Jinxiang Chai, and Jessica Hodgins. 2015. Realtime Style Transfer for Unlabeled Heterogeneous Human Motion. *ACM Trans. Graph.* 34, 4, Article 119 (July 2015), 10 pages.

M. Ersin Yumer and Niloy J. Mitra. 2016. Spectral Style Transfer for Human Motion Between Independent Actions. *ACM Trans. Graph.* 35, 4, Article 137 (July 2016), 8 pages.