Automated Choreography Synthesis Using a Gaussian Process Leveraging Consumer-Generated Dance Motions

Satoru Fukayama, Masataka Goto
National Institute of Advanced Industrial Science and Technology (AIST)
1-1-1 Umezono, Tsukuba, Ibaraki, Japan
{s.fukayama, m.goto} [at] aist.go.jp

ABSTRACT

We propose a novel method of automatically generating dance choreography using machine learning. In a typical approach to automatic choreography, a dance is constructed by concatenating segments of existing dances which are maximally correlated to the target audio features with connectivity constraints. However, researchers using this approach are unable to produce dances with much variety, since the set of examples used in these experiments (usually motion-capture of existing choreographies) is limited and costly to produce. To solve this issue, we propose a probabilistic model which maps beat structures to dance movements using a Gaussian process, trained with a large amount of consumer-generated dance motion obtained from the web. The main contribution of our work is the combination of two approaches: the previously mentioned correlation based approach which seeks for relationships between music and dance, and a machine learning approach which is based on human motion modeling. Inspection of the generated dances proves that our method can generate choreographies with different characters by switching the training dataset, and highlights opportunities in training with further dance motions on the web to generate more expressive dance choreography.

Author Keywords

Automated Choreography; 3D-animations; Consumer-Generated Media; Gaussian Process

ACM Classification Keywords

I.3.8 J.5

INTRODUCTION

Technologies for rendering dances of a three-dimensional animation character now enable ordinary viewers of video clips on the Internet to create their own original animations. Software such as Maya[1] from AUTODESK[2] supports the creation of visual effects, games and animations. Even without technical knowledge, free software, such as MikuMikuDance (MMD), allows users to control the movement of various three-dimensional characters sourced from the Internet.

ACE '14, November 11 - 14 2014, Funchal, Portugal

ACM 978-1-4503-2945-3/14/11$15.00
http://dx.doi.org/10.1145/2663806.2663849

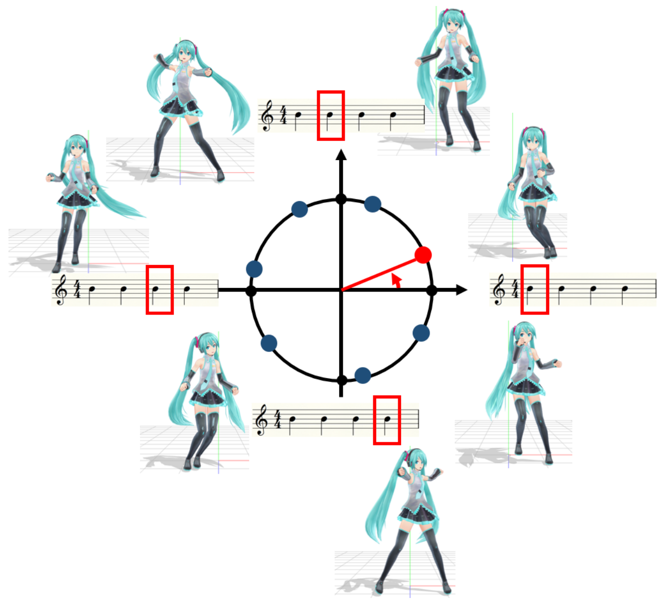

Figure 1. Beat structure is modeled over a circular trajectory, with period equal to the time in a musical bar. The relationship between the beat structure and motion is modeled mathematically with a non-linear transform. The name of the computer-graphics character used in this figure is *Hatsune Miku* and its 3D model is created by koron.

We are studying automated choreography since creating dance motion through current methods is time consuming and difficult. Creating dances by hand often involves frame-by-frame editing of individual body parts. This is clearly cumbersome, and can be done only by experts. Using motion capture data partially resolves this problem, but only who can dance can utilize it to create dance motion and not available to non-professionals.

There is a great potential for leveraging a large amount of consumer-generated choreographies for analyzing and syn-

[1] `http://www.autodesk.com/products/autodesk-maya/overview`
[2] `http://www.autodesk.com/`

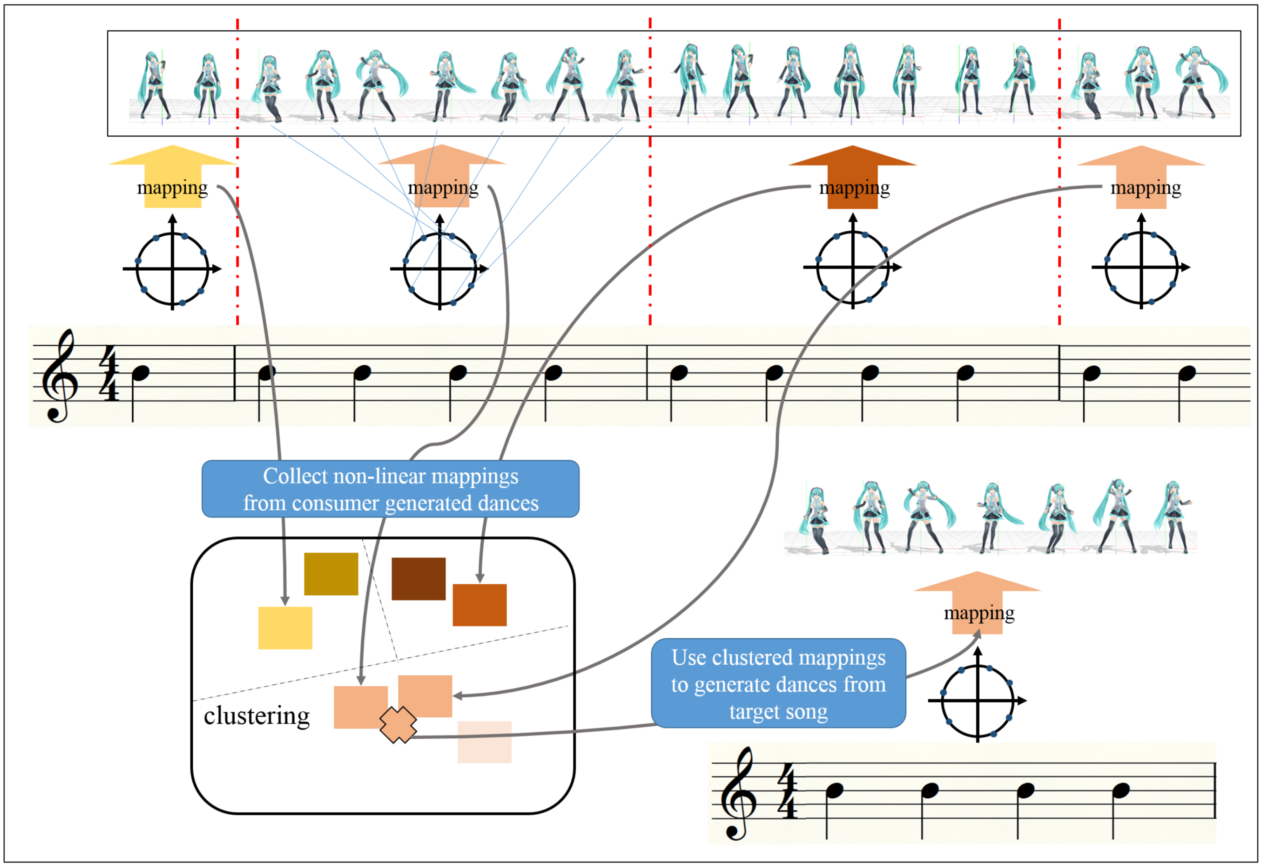

Figure 2. Overview of our method.

thesizing dance motions. For example, on the most popular Japanese video sharing service, *Niconico*[3], we find an enormous number of 3D-animation videos created using MMD, the vast majority of which are dances [6]. The availability of this vast quantity of dance data enables a potential paradigm shift in automatic choreography towards machine learning-based approaches, since the model for generating dance motion may now be trained or inferred in a data-driven manner.

In this paper, we propose a method for synthesizing dance choreography based on an analysis of consumer-generated dance motions found on the web. In contrast to existing techniques, our method is able to leverage readily available rich online content, and also can provide creators with insight into the general dance moves which are currently popular online.

Relevant work

Various researchers have attempted to generate dance motion from music [4], the dominant approach being attempting to uncover relationships between existing dances and their accompanying music. Music content such as tempo [1, 16], acoustic beat features [17, 9], pitch and chord data [14], and the melodic contour [15] have been used to analyze these relationships, in addition to combinations of acoustic features [3] and structural similarity [10].

Another topic investigated in this area is the mathematical modeling of realistic human motions. The use of non-linear mapping from (hierarchical) latent variables to dance motion is the common solution applied in this body of research. Hidden Markov Models (HMM) [19], Dynamic Bayesian Networks (DBNs) [12] and Hierarchical Dirichlet Process-Hidden Markov Models (HDP-HMM) [20] have been applied to model dance steps or human motion in general. There are also methods which emphasize the abstraction of the posture information of dances by using Kernel Canonical Correlation Analysis (KCAA) [7], Gaussian Process Dynamical Models (GPDM) [21], Topological Gesture Analysis (TGA) [13, 18] or Multi-layer Joint Gait-Pose Manifolds (multi-layer JGPM) [2] to represent dance motion in a compact two-dimensional space.

Novelty of this work

We aim to design a probabilistic model that handles both relationship between music and dance motion and also the non-linear mapping from latent variable trajectory to dance motion, leveraging consumer-generated choreographies to train the model. Our key contributions are:

- the mapping of musical content to dance motion using non-linear transforms,

[3] `http://nicovideo.jp`

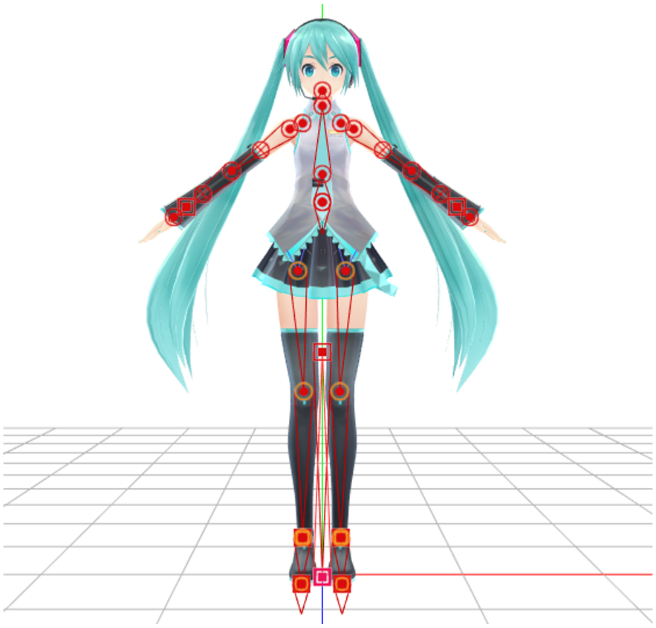

Figure 3. Joints used to analyze and generate dance motions are shown as red circles on a character. Other joints can automatically move based on inverse kinematics to follow the joint motions generated by our method.

- the clustering of probabilistic models representing the existing rich data available from consumer-generated choreographies, and
- the generation of new choreographies using the power of this clustered model.

The novelty of our method lies not only in the use of consumer-generated dance motion but also on how the above problems can be solved. Taking the view that the beat information is most important for creating dance motion, we begin by modeling the beat structure over a circular trajectory, with period equal to the time in a musical bar (Fig. 1). The non-linear mapping between the circular trajectory and the dance motion is modeled mathematically with a Gaussian process, which facilitates the learning of dance characteristics using maximum-likelihood estimation from training data. After non-linear mappings are obtained from every bar in the training data, they are clustered into an optimal number of clusters using a Bayesian Information Criteria-based k-means algorithm. Finally, dance motion for target music is generated by concatenating clusters.

METHOD FOR AUTOMATED CHOREOGRAPHY LEVERAGING CONSUMER-GENERATED DANCES

An outline of our approach in this work is shown in Fig. 2. The first step in our method is to capture relationships between beat structures and choreographies. We split dance motion into segments defined by bar lines of the accompanying music. The bar lines are determined by using an existing automatic beat analysis method for music audio signals. In each bar, we aim to infer a non-linear transform which maps beat structure to dance motion. The beat structure in a bar can be represented by a circular trajectory where the time period is equal to the duration of a bar in seconds. Inferring the relationship between this circular trajectory and the dance motion is formalized mathematically with a Gaussian process. As a result, we obtain a probabilistic model for each bar, which holds information regarding the mapping between the beat structure and the dance motion (see top of Fig. 2).

Next, after obtaining many probabilistic models (non-linear mappings) from consumer-generated dance motions for various songs, we create clusters of probabilistic models so that dance motions generated from the probabilistic models are similar within each cluster. Our method simultaneously optimizes the number of clusters in a way designed to limit overfitting. We obtain typical dances observed in consumer-generated choreographies which can be used to generate new dance motions (see the bottom left of Fig. 2).

Finally, we generate dance motions for a new song (the 'target song') by automatically selecting appropriate clusters of the probabilistic models. Each bar in the beat structure obtained from the beat analysis of the target song defines the latent circular trajectory. A cluster obtained in the previous clustering step is then automatically selected for each trajectory by considering the connectivity with adjacent bars. A user can also manually select a new cluster if they are displeased with the current dance motion. The probabilistic model of the selected cluster finally generates dance motions corresponding to the trajectory (see the bottom right of Fig. 2). In concatenating two different models, we linearly interpolate body motions between the current motion and the following motion during the fourth beat of the bar to avoid the gap between motions. This interpolation is done during the fourth beat to preserve the starting motion on the strongest first beat of the bar.

Representing dance motion

The dance motion of a human body can be represented by positions and rotations of movable joints. We followed the convention used in MMD and work with 20 joints shown in Fig. 3. The movement of each joint can be described using position values and quaternions. By collecting all position values and quaternions into a vector, we arrive at an 81-dimensional vector representing the pose at each time of a dance motion. Let $\mathbf{d}_t$ be this vector for representing the dancer's pose at time t.

Representing beat structure

In popular dance music, beat structures are mostly periodic. For example, it is most common to have music in 4/4 meter, with the strongest beat attacks occurring on beats 1 and 3, with little or no change throughout the song. Within a bar, beats also have a symmetric structure. For instance, a down-beat usually follows an up-beat and vice-versa.

Sine waves or sequences of delta functions in one dimension are also periodic and seems to represent the beat structure, but they do not hold symmetric properties to represent the different functions of a beat such as the up-beat and down-beat in a bar.

To represent periodic and symmetric properties of a beat structure, we choose a circular trajectory cycles around with period τ, equal to the time duration of a bar. Let t_{s} be the time where a bar starts, and $\mathbf{b}_t$ be the representation of beat structure at time t. $\mathbf{b}_t$ is a two-dimensional vector:

$$\mathbf{b}_t = \left(\sin\left(2\pi\left(t - t_{\mathrm{s}}\right)/\tau\right), \cos\left(2\pi\left(t - t_{\mathrm{s}}\right)/\tau\right)\right).$$

The circular trajectory can represent both the symmetric and periodic structure of the beats.

Non-linear transform from beat to motion

Our goal has been to construct a model which describes the relationships between the sequence of dance motion $\mathbf{D}$ and the beat structure representation $\mathbf{B}$:

$$\mathbf{D}_{1:T} = \left[\mathbf{d}_1 \cdots \mathbf{d}_T\right]^{\mathrm{T}}, \ \mathbf{B}_{1:T} = \left[\mathbf{b}_1 \cdots \mathbf{b}_T\right]^{\mathrm{T}},$$

where T is the total duration of the target song for generating dance motion. Since we set the beat structure representation to be periodic, we can divide our solution into two steps: first, constructing a model within a bar ($t_{\mathrm{s}} \leq t \leq t_{\mathrm{s}} + \tau$), and second, constructing a combined model for the entire duration of the song ($0 \leq t \leq T$).

We construct a probabilistic model which describes the most probable dance motion given the training data and beat structure. If we think that dance motion is generated from a circular beat structure trajectory, this motion can be described by a non-linear transform $f(\cdot)$ as: $\mathbf{d}_t = f(\mathbf{b}_t)$. It is perhaps useful to think of f functioning analogously to the gears in a motorized vehicle, converting a circular trajectory (the piston) into the desired forward motion. By assuming that a Gaussian noise (with variance σ^2) is added for observing the dance motion as $\mathbf{d}_t = f(\mathbf{b}_t) + \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}\right)$, and each $\mathbf{d}_t$ at $t_{\mathrm{s}} \leq t \leq t_{\mathrm{s}} + \tau$ is generated independently from $\mathbf{b}_t$, the probability for observing a given dance in a single bar generated from beat structure is:

$$p\left(\mathbf{D}_{t_{\mathrm{s}}:t_{\mathrm{s}}+\tau} | \mathbf{B}_{t_{\mathrm{s}}:t_{\mathrm{s}}+\tau}\right) = \prod_{t=t_{\mathrm{s}}}^{t_{\mathrm{s}}+\tau} \mathcal{N}\left(f(\mathbf{b}_t), \sigma^2\mathbf{I}\right). \qquad (1)$$

Inference of transform using a Gaussian process

Inferring $f(\cdot)$ from training data can be done using a Gaussian process. We briefly review the logic for inferring $f(\cdot)$.

Given the training data, we wish to optimize $f(\cdot)$ in order to achieve maximum log-likelihood. Representer theory shows that the optimal $f^*(\mathbf{b}_*)$ which can be used to predict a new $\mathbf{d}_*$ from $\mathbf{b}_*$ is a linear combination of τ kernel functions, each one defined on a training point as:

$$f^*(\mathbf{b}_*) = \sum_{t=t_s}^{t_s+\tau} \alpha_t k(\mathbf{b}_t, \mathbf{b}_*),$$

where $k(\mathbf{b}_t, \mathbf{b}_*)$ is a kernel function. In our research, we followed the convention of using squared exponential covariance function as kernel function:

$$k(\mathbf{b}_t, \mathbf{b}_*) = \exp\left(-\frac{\lambda^2}{2}|\mathbf{b}_t - \mathbf{b}_*|^2\right), \qquad (2)$$

Furthermore, Mercer's theorem shows that:

$$k(\mathbf{b}_t, \mathbf{b}_*) = \sum_{i=1}^{\infty} s_i f_i(\mathbf{b}_t) f_i(\mathbf{b}_*), \qquad (3)$$

where s_i are the eigenvalues for the function f. By applying Eq. (2) and (3) to Eq. (1), and marginalizing over α_i by the assumption of $(\alpha_1 \cdots \alpha_{t_{\mathrm{s}}}) \sim \mathcal{N}\left(\mathbf{0}, \lambda^{-2}\mathbf{I}\right)$, we formulate the probability distribution which represents the relationships between the dance motion and beat structure:

$$\begin{aligned} & p\left(\mathbf{D}_{t_{\mathrm{s}}:t_{\mathrm{s}}+\tau} | \mathbf{B}_{t_{\mathrm{s}}:t_{\mathrm{s}}+\tau}, \sigma, \lambda\right) \\ = & \frac{1}{\sqrt{(2\pi)^{\tau d}|\mathbf{K}|^d}} \exp\left(-\frac{1}{2}\mathbf{D}_{t_{\mathrm{s}}:t_{\mathrm{s}}+\tau}^{\mathrm{T}} \mathbf{K}^{-1} \mathbf{D}_{t_{\mathrm{s}}:t_{\mathrm{s}}+\tau}\right), \end{aligned}$$

where $d = 81$ for the dimension of the dance motion data, and $\mathbf{K}$ is the kernel matrix defined by:

$$(\mathbf{K})_{ij} = k\left(\mathbf{b}_{t_s+(i-1)}, \mathbf{b}_{t_s+(j-1)}\right) + \sigma^{-2}\delta_{\mathbf{b}_{t_{\mathrm{s}}+(i-1)}\mathbf{b}_{t_{\mathrm{s}}+(j-1)}},$$

where $\delta_{\mathbf{b}\tilde{\mathbf{b}}}$ is a Kronecker delta, equal to one if $\mathbf{b} = \tilde{\mathbf{b}}$ and zero otherwise. Note that the variance σ^2 for observing the dance motion is added to the kernel function. The parameters (σ, λ) can be optimized by maximizing the log-likelihood with respect to the parameters with the scaled conjugate gradient method [11].

With this probabilistic model and the optimized parameters (σ^*, λ^*), we transform the new latent circular trajectory $\mathbf{B}^*_{t^{‘}:t^{‘}+\tau^{‘}}$ to dance $\mathbf{D}^*_{t^{‘}:t^{‘}+\tau^{‘}}$ by:

$$\mathbf{D}^*_{t^{‘}:t^{‘}+\tau^{‘}} = \mathbf{K}^* K^{-1} \mathbf{D}_{t_{\mathrm{s}}:t_{\mathrm{s}}+\tau}$$

where $\mathbf{K}^*$ is a kernel matrix defined as:

$$(\mathbf{K})_{ij} = k\left(\mathbf{b}^*{}_{t^{‘}+(i-1)}, \mathbf{b}_{t_s+(j-1)}\right).$$

Clustering the probabilistic models

Given a collection of probabilistic models generated from song measures, we now wish to create clusters and obtain representative dances by calculating the centroids of these clusters. We can also optimize the number of clusters by considering the variety of dance motions.

To choose the number of clusters, we can use an information criterion. Since the likelihood is at its maximum when the number of clusters is equal to the number of data points, we need to put a constraint on increasing the number of clusters and to avoid overfitting the data. The Bayesian Information Criterion (BIC) is often used to determine how much a model is overfitting. Specifically, BIC may be calculated as:

$$\mathrm{BIC} = -2\sum_{k=1}^{K} L_k + Kd\log(N) \qquad (4)$$

where K is the number of clusters ($k = 1, \cdots, K$) and N is the number of collected probabilistic models with a Gaussian process. L_k is the log-likelihood of cluster k with an assumption that data points in a cluster are Gaussian distributed.

K-means can be combined with BIC to optimize the number of clusters. By starting from a single cluster where the mean is equal to the mean considering all models, we split the cluster with the maximum variance and keep splitting till the BIC value starts to increase.

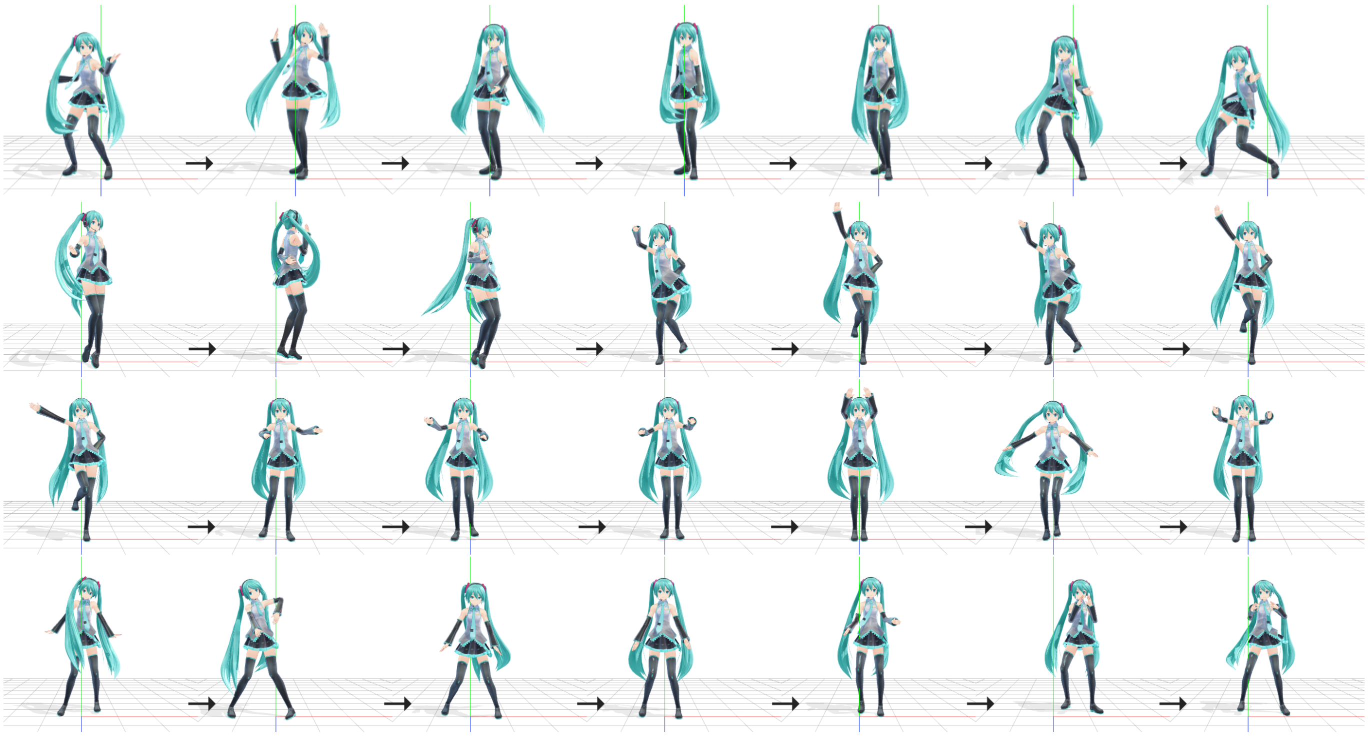

Figure 4. Example results of generated dance motion.

Generation of dances from a target song
To generate the dance motions, we start by randomly selecting one cluster centroid, and then generate a sequence of clusters by considering the connectivity constraints. To generate smooth intersection between adjacent clusters, we apply linear interpolation between the current dance and the following dance on the fourth beat. In our current implementation, the generated output is in *vmd format* and can be rendered into computer graphics by using MikuMikuDance software. Since the generated motion data holds general information to specify human poses, dances can also be rendered by other software such as Maya.

EXPERIMENTAL RESULTS

Training data and results
We used choreography data which consisted of around 640,000 human posture data points and used them to train our model. The data consists of position values and quaternions of the dance motion. Beats were automatically detected from audio signals of each song. However, for the purposes of evaluation in this pilot study, we manually corrected beat detection errors. We used RWC-MDB-P-2001 No. 7 from the RWC Music Database [5] as target song to generate dance. Example dances automatically generated from different clusters of probabilistic models are shown in Fig. 4. The name of the computer-graphics character used in this figure is *Hatsune Miku* [8] and its 3D model was created by koron.

Discussion
The dances generated with different clusters had the characteristic motion of each cluster, which indicates we could obtain more variations by leveraging more consumer-generated choreographies. Although our approach can extract periodic dance motion where the period time is a bar length, some dance motions in the database are originally designed to have longer periods. By constructing several probabilistic models regarding different period times and combining these in linear combination, we will be able to represent more complicated dances.

We also tried and found that impressions given by dance motions can be controlled by dynamically changing the speed of circular moves along the latent circular trajectory. This speed is constant in usual, but when we slowed it down right after every beat and speeded it up right before every beat, for example, impressions became zippier and more energetic. This has great potential as a way to easily customize dance impressions.

CONCLUSIONS AND FUTURE WORK
We have developed a novel method of automated dance choreography synthesis which leverages consumer-generated dance motion. Experimental results showed that dance motion database can be leveraged to generate choreographies in currently popular styles.

Our current implementation only focuses on the beat detection results, whereas other musical features such as chord progressions, mood, melody lines and lyrics could also be informative in the automatic creation of dance choreographies. As our next step, we plan to extend our method to integrate more music features for analyzing and synthesizing dance motions.

ACKNOWLEDGEMENT

Hatsune Miku is copyrighted by Cripton Future Media INC., and we used it in our research under PIAPRO license (`www.piapro.net`). 3D model of *Hatsune Miku* used in our research is created by koron. This work was supported in part by OngaCREST, CREST, JST.

REFERENCES

1. Chen, K. M., Shen, S. T., and Prior, S. D. Using music and motion analysis to construct 3D animations and visualisations. *Digital Creativity 19*, 2 (2008).
2. Ding, M., and Fan, G. Multi-layer joint gait-pose manifold for human motion modeling. In *Proc. FG 2013* (2013), 1–8.
3. Fan, R., Xu, S., and Geng, W. Example-based automatic music-driven conventional dance motion synthesis. *IEEE Transactions on Visualization and Computer Graphics 18*, 3 (2012).
4. Goto, M. An audio-based real-time beat tracking system for music with or without drum-sounds. *Journal of New Music Research 30*, 2 (2001), 159–171.
5. Goto, M. Development of the RWC Music Database. In *Proc. ICA 2004* (2004), 553–556.
6. Hamasaki, M., and Goto, M. Songrium: A music browsing assistance service based on visualization of massive open collaboration within music content creation community. In *Proc. WSOS 2013* (2013).
7. Hirose, T., and Taniguchi, T. Abstraction multimodal low-dimensional representation from high-dimensional posture information and visual images. *Journal of Robotics and Mechatronics 25*, 1 (2013).
8. Kenmochi, H. VOCALOID and Hatsune Miku phenomenon in Japan. In *Proc. InterSinging 2010* (2010), 1–4.
9. Kim, J. W., Fouad, H., Sibert, J. L., and Hahn, J. K. Perceptually motivated automatic dance motion generation for music. *Computer Animation and Virtual Worlds 2009 20* (2009), 375–384.
10. Lee, M., Lee, L., and Park, J. Music similarity-based approach to generating dance motion sequence. *Multimedia Tools and Applications 62*, 3 (2013), 895–912.
11. Møller, M. F. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks 6*, 4 (1993), 525–533.
12. Moon, K., and Pavlović, V. Graphical models for human motion modelling. *Human Motion, Computational Imaging and Vision 36* (2008), 159–183.
13. Naveda, L., and Leman, M. The spationtemporal representation of dance and music gestures using topological gesture analysis (TGA). *Music Perception 28*, 1 (2010), 93–111.
14. Ofli, F., Erzin, E., Yemez, Y., and Tekalp, A. M. Learn2dance: Learning statistical music-to-dance mappings for choreography synthesis. *IEEE Transactions on Multimedia 14*, 3 (2012).
15. Oore, S., and Akiyama, Y. Learning to synthesize arm motion to music by example. In *Proc. WSCG 2006* (2006), 201–208.
16. Panagiotakis, C., Holzapfel, A., Michel, D., and Argyros, A. A. Beat synchronous dance animation based on visual analysis of human motion and audio analysis of music tempo. In *Proc. ISVC 2013* (2013), 118–127.
17. Shiratori, T., Nakazawa, A., and Ikeuchi, K. Synthesizing dance performance using musical and motion features. In *Proc. ICRA 2006* (2006), 3654–3659.
18. Sousa, P., Oliveira, J. L., Reis, L. P., and Gouyon, F. Humanized robot dancing: Humanoid motion retargeting based in a metrical representation of human dance styles. In *Proc. EPIA 2011* (2011), 392–406.
19. Takeda, T., Hirata, Y., and Kosuge, K. Dance step estimation method based on HMM for dance partner robot. *IEEE Transactions on Industrial Electronics 54*, 2 (2007).
20. Taniguchi, T., Hamahata, K., and Iwahashi, N. Unsupervised segmentation of human motion data using sticky HDP-HMM and MDL-based chunking method for imitation learning. *Advanced Robotics 25*, 17 (2011).
21. Wang, J. M., Fleet, D. J., and Hertzmann, A. Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Recognition and Machine Intelligence 30*, 2 (2008).