Research on Refinement and Real-time Editing of Expression for 3DCG Character Animation

3DCG キャラクタの表現の改善法と実時間操作に関する研究

February 2019

Takuya Kato

加藤 卓哉

Research on Refinement and Real-time Editing of Expression for 3DCG Character Animation

3DCG キャラクタの表現の改善法と実時間操作に関する研究

February 2019

Waseda University
School of Advanced Science and Engineering
Department of Pure and Applied Physics, Research on Image Processing

Takuya Kato

加藤 卓哉

Contents

To my wife and my family

Chapter 1

Introduction

1.1 Background

The ways to enjoy music have transformed dramatically over the few decades. With the introduction of the internet and streaming services, the accessibility of the the music has been improved that people can enjoy the music anywhere and anytime. From the artists stand of point, the software to help compose and edit the music using the computer has been one of the popular choice to make when creating the music. The artists are now allowed to share the music that they created through the internet for many people to listen to. From the listeners point of the view, it has been much easier for them to find and listen to the music on the internet and share them with people who might be interested. The listeners are even give a comment or create secondary contents using the music to create new contents. Such changes in the field have altered the role of the visual contents using the music. The visual contents using the music has long history from the plays, operas, musicals, films until the music videos. These transitions took the contents outside of the stages and allow the viewers to enjoy them without going out for them. With the help of the TV and computers, music video has become one of the best ways to distribute the music effectively. Through the music videos, the musicians are benefited not only to be able to distribute their contents but also economically and becoming more popular. With the introduction of various technologies, the music videos have evolved into many different contents. One of the genre of the music video that is growing exponentially is the musical animations. The musical animations is the animation that matches to the music by character singing or dancing. Using animation instead of the live actors or the actual videos allows the musician to be able to create literally anything within the computer generated world. With the development of the computer graphics

software and tools, the threshold of creating the animation has lowered, that many artists are beginning to create their own computer generated character and creating the video that these avatars singing and dancing to music. The trend of the musical animation made the impact to the world by creating one of the ultimate contents of computer graphics and the musical animation industry; the live performances of computer generated avatar. The growth of the streaming services, the value of prerendered and not interactive media contents has regressed and the value of the interactiveness in live performance have reacknowledged. The CG avatar on the stage dances and sings to the music for people to enjoy requires tedious work of both artistry and technology. In order to make live stage performance successful, not just the representation of an avatar should be extremely realistic, but also the movement of the character requires to be realistic. Additionally, interacting with the real human on and off the stage is required in order to make the live performance valuable than the prerendered music video. By accomplishing such technical difficulties, many believe that we are able to realize the live stages such as reliving the artists that have passed away in the past or creating fully computer generated pop-stars.

1.2 Research Goal

The goal of the researches in this dissertation is to provide the tools to support creating musical animation, which essentially leads to creating the live stage performances. The field of the musical animation locates in between the music information processing and character animation. Some of the techniques and ideas can be shared in common from both of the fields, yet many aspects of the procedures, ideas or the technologies are different from either of the fields. The evidence of such is stated in the research field in dance motion. Since dance motion is isolated and unique from the daily motion, the formation of the research field is very unique. Many of the dance motion synthesis technique requires both the information of the music to be defined as the dance motion, while the motion itself follows the physical constraint of the character animation. The balance of the entertainment, realism and physicality of the character is essential, and these aspects requires to be studied independently from each fields. Since the field of musical animation is not established as of yet,

many new concepts as well as data configuration needs to be implemented to consider the relationship of the motion, music and the final animations to be rendered. As much as the result of the created animation itself is important, understanding correlations of the variables through its creation is essential. Through the research in musical animation creation, the glimpse of the correlations between the musical information and character animation will hope to be revealed which optimizes the process of the creation in the future.

1.3 Focus of the Dissertation

The process of creating the musical animation disciplines into four layers; the modelling phase, animating phase, rendering phase and realtime control phase. Among all, there are two major aspect of the musical animation in this dissertation. The common factors of these topics are the fact that these two areas particularly require technique that specified in musical animation compared with other aspects such as modelling, body motion or rendering. Other areas requires the problem in music, while there are some fields that requires musical information to be taken in to consideration. This dissertation will be focusing on the topics that are specific for musical animation that would help the musical animation industry immediately. One of the topics that this dissertation focuses is the facial animation, especially when the avatar is singing. The facial expression while singing have three major features. Firstly, they are different from the daily facial expressions. While many speech animation creation research have been developed, the behavior in speech animation and that of the singing animation differs as the dance motion is different from the daily human motion. The other feature is that the expressions are known to corresponds to the singing voices. Previous studies mentioned mentioned about the fact that the audience be able to plot the height of the musical tone only looking at the facial expression and head rotation. Lastly the expressions in singing is unique to the individuals, the facial expression cannot be identical between the characters. This indicates that the generalizing the motion that can be applied to everyone will not satisfy the quality of the viewers. By adding facial expressions to live stage performance, the realism of the animation will increase significantly. The other area that

this dissertation focuses is the interface and the way to interact with the musical animation avatar in realtime. There are existing interfaces and the interactions research in order to contorl virtual character in games, VR or AR, the interfaces specialized in musical animation control in realtime does not exist. The control of the character in realtime specifically for the musical animation requires the interface to be able to consider the musical aspect of the character controls. In such cases the interface that is already been used in musical editing could be transferred.

1.4 Dissertation Organization

In this dissertation, I will be proposing three animation synthesis research that helps the artist to edit the expressivity of the character. There will be three different layers in this dissertation on the editing of the expressivity; Creating the basis model for each of the character, setting the animation blending weight of the basis model of the character and controlling them by understanding the features of the characters. This dissertation covers this three layers by focusing on the fact that the rough sculpting or the input that derives the animation can be acquired. From those inputs, the sufficient details will be estimated from the training data and applied to the roughly sculpted models for musical animation production. These are the 3 research in this dissertation;

Basis Geometry Sculpting (Chapter 3): To create characters, one of the most simple ways to create them is to set a geometry a frame by a frame. To make the animation creation process efficiently, we are able to use blendshape animation, which is the technique to linearly add multiple facial geometries and set the adding weight to create the animation. Our first work will be presenting the way to create the basis model of blendshape. The synthesize results created by using this work is successfully able to add expressivity to the roughly sculpted basis model created by the previous work by modelling the mapping of the characters tendency of moving their body or face. Since the blendshape is one of the most popular ways to create character animation, the proposed method fits into animation creation process smoothly to help create a animation.

Blend Weight Setting (Chapter 4): Once the basis expression or the pose is set, we apply the animation weight to the character to control the character. While the parameters are only limited to the numbers of the basis, the animators are required asked to control every single frames of in the key frames, it is one of the most important yet difficult task they are asked to do. I will propose the way to automatically generate the blending weight of each frame just from the voice of the actor. This will allow the animator to get the basic yet the detailed facial expressions corresponded to the actors voice. Using the methods in chapter 3 and chapter 4, animators will only required to create few facial model to create basis expression and set the blending weight.

Real-time editing of the expressivity in animation (Chapter 5): While many 3DCG characters are controlled on offline, demand in real-time controls of the characters are getting higher and higher. In offline process, the animators can accepts the fact that the automatically created animations be wrong in some sense, while in real-time control it will not be allowed. It is required for the animators to know whether the character is with =in the expressivity that the characters had been defined. This dissertation includes the way to how we can visually teach the animators how much the synthesized results went far from the characters expressivity space. By introducing novel interface and design the user interface to see how far the created animations went from the center of the character's expressivity space, the animators will be able to control the character smoothly.

These three layers for each of the process will be able to help the artists to focus on how to create one single scene, so that the other scenes can be created by reusing the models and animated results. In this dissertation, I will first discuss the animation creation pipeline to provide the basic ideas of how to create animation. Then following chapters will be explaining each of the topic above.

Chapter 2

Character-Animation Pipeline and Controls

Character animation process is separated into three major procedures —modeling, rigging and animation. Creating animation from the scratch requires starting the process all over again for specific to each character. The research presented in this dissertation aims to provide a degree of compatibility in the animation to control the character animation especially in musical animation. My works not only considers the novelty of the ideas and application, but also considers the fact that it will inserted into practical animation creation procedures. In order to consider , these contributions are best understood in the context of current animation procedures. This chapter presents the four primary stages in the musical animation pipeline. Each stage have a different set of goals and challenges. I describe what kind of the technique exits in each process and what kind of technique is demanded in order to ease the process of creating the musical animation.

2.1 The process of creating character animation

2.1.1 Modeling

The initial step in creating animation is to generate a 3D geometry of the character in order to be animated. This process has various names such as modeling, mesh editing or sculpting, and a numerous ways are used to achieve it. In 3DCG creation, modelling a character using 3D animation tools has been common, as well as capturing the geometry from the real world object. Especially in the case of modelling the real human facial animation, the facial scans are used to create not only the model of the neutral face of the model. Numerous tools for creating the model is introduced, the accessibility in terms of the price and the size of the scanners is rising. Research trend of the real human modelling has been highly demanded with the growth of

varieties of applications, reconstruction of the high quality model from a portrait is introduced. For non-photorealistic character, 3DCG creation tools have been updated day-by-day and it's creation has also been accessible. The scanning of the real world clay or the photo is not new to the research field and less works are required for the artists to model or edit the character.

In the musical animation, this process is no different from the modelling of the other 3DCG animations. Many technique can be transferred from the Computer Graphics field, and has been successfully used in many of the existing musical animation field.

2.1.2 Rigging

After the representation of the character has been created, instrumenting a character with the sets of controllers that offers the artists to deform the character to various poses. The technique of modeling is not always appropriate for animation since the numerical representation employed by modeling tools does not encode the animator's artistry well on how the bosy of the character are made to deform. The set of semantic deformations that is restricted in certain areas of the bosy determines the kinematics of the mesh, or how the vertices of the body meshes are set to be controlled. The character kinematics includes the animator's high-level understanding on which body deformations are appropriate for the mesh in question. This process is called rigging to be different from modelling, since modeling process does not have a control on mesh kinematics. There are number of choices to be made on which rigs to be used, the common rigs that are used is the pose space deformations. In pose space deformations, few base poses are defined and the character will deform by the linear combination of the base poses. One of the most popular pose space deformation example is the blendshape animation in facial animation. According to Ekman and Rosenberg, 1997, facial expression taxonomized into several facial movements by their appearance on the face called Facial Action Coding System, FACS in short. Such representation of categorizing the facial expression is widely spread and many facial animation rigs are based on the FACS as the basis of the blendshape animation rigs. From the past to the present, the popular pick to

used for rig representation for the whole body is based on bone animations. Compared to facial animations, the degree of freedom on full body animation is large and complex. By using bone animation, the animators are able to control the character with more flexibility and controlability. In this dissertation, we follow main stream represention for both the facial animation and the full body animation; using blendshape animation for facial animation and bone animation for full-body animation. The importance in choosing which rig to use should depend on the popularity and the trend among artists, and the impact of the research will be more significant by following them.

2.1.3 Animation

Once modelling and rigging has been allocated to the character, animation can be applied. Animating a character is the process which sets the numerical values for the rigs in order to control the character over the time frame. The values are set by using keyframes in which the controls are set at certain frames which is thought to be important comapared with other frames. Other frames which is not the keyframes are interpolated over the time frame with certain interpolation . Keyframe animation provides the creative control for artists to control over the animation but the artist are required to generate set keyframes with high volume and with details. If more realistic animation is requires, the motion of a real actor will be captured and transferred. Motion capture system can acquire bone motion that can be transferred to the character using the skeleton-based rig methods. Physics-based animation methods, that simulates the physically appropriate simulation models to control the character's movement, is able to create super-realistic animation with in the certain constraints of physics. On the flip side, the level of difficulty in order to control high-level physics simulations can cause problems when the artists are required to create certain effect that overturns the physical constraints.

In animation, the ways to control them is plays significantly large role. Setting keyframe for animation can be done in animation that allows the offline rendering, but it cannot be possible in the case of animation rendered in realtime. In this case the interface to control the animation is fully corresponds to the final quality of the

animation. There are various controling methods in character animation such as motion capture or gaming controllers, these controllers have the bottleneck in different occasions. The motion capture can be used in both realtime and offline animation for high quality body motion creation. The motion captureing system has also been accessible to public in various ways, many characters are currently controlled by motion capture. The bottle neck of the motion capture is that the actors are required to be able to do the same action that the artist wish the character to do. There are many cases that the actor is not accessible or the actor cannot do the required motions, such as the high level dancing, the alternative solution is required to be considered. The gaming controllers are better solutions in such cases, especially in the case where the animation is restricted to simple movements. Once the motion is set, the motion can be relive with the press of buttons. The interfaces to control the character is being studied especially in VR and AR field recent years.

2.2 The Requirements for musical animation

The layers of creating the animation is common for majority of the animation industry, the problem that each kinds of animation have differs depending on what needs to accomplished and the importance that viewer weigh on. In the case of musical animation, it differs from other fields more since the centroid of the animation is on the music and animation follows the music. Expect for the special circumstances, the order does not flip in the process. Considering this fact, the process of musical animation creation is categorized into the process that requires to consider music or not. As for modelling and rendering, music does not play a major roles in terms of decision making. In fact, the requirements in these process is common between other computer graphics animation. Accordingly, these fields requires can be considered to be outside of musical animation research field. What we need to consider more in musical animation is rigging and animation phases. Therefore, in this dissertation, we have looked into the research on creating musical animation by focusing on rigging and animation. By using the techniques introduced in other animation fields for modelling and rendering, the musical animation creation can be optimized and the creating them would be efficient and accessible to many. In the following

chapters, we introduce the research on the rigging and animation phases. Not only the research follows the current scheme of the current CG creation process, but also introduce the new concepts and ideas to be inserted.

Chapter 3

Example-based individuality retargeting for facial animations

3.1 Introduction

It is known that facial expressions of characters play larger roles when creating a animation. As being addressed in previous chapter, blendshape animation enables the artists to control the face to generate various expressions for characters by linear combination of a certain basis of expressions models known as blendshapes. Although, there lies a challenge in which the artists are required to control the blending coefficients parameters and model the blendshapes to achieve highly realistic animation. Some research has introduced methods to estimate the blending coefficients automatically using the various techniques such as deep learning. While such methods provides artists to be able for them to create roughly created animations with efficiency, it is still a difficult task to create ideal expressions only with the set of parameters in blending coefficients. Therefor, artists are required to build an various high quality blendshapes models that may have similarity with other blendshapes but to be used in different scene with in the contents. In order to address the definition of problem, modelling blendshapes with character-specified expressions remains a intensive labor in the process since the expressions in between the character differs greatly. Not only the semantics of the expressions, individualities in character expressions are diverse. For instance, when the topologies of the characters are completely different the ways to laugh will be significantly different because each character moves the facial parts in a completely different way. Therefore it is very time-consuming procedure to build model of numerous number of blendshapes for each individual characters when creating blendshapes for various characters that appears in the contents.

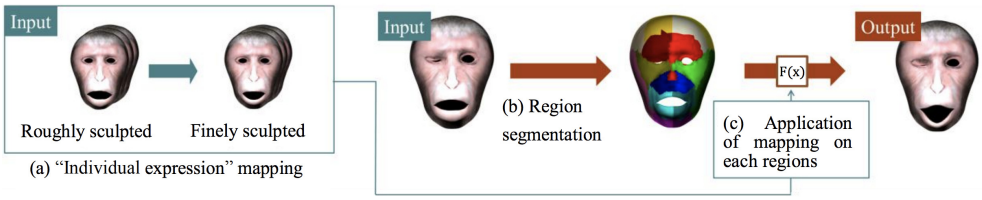


FIGURE 3.1: Overview of "Character Transfer" that (a) creates mappings using training examples. (b) Automatically segments the region, and (c) applies blended mappings onto each segment region.

In this chapter, we propose a technique called "Character Transfer" that refines the automatically created rough input blendshapes to model more detailed and specified expressions for the character. This methods allows the artists to model the individualities by a small number of training examples on automatically clustered regions of the animation. The contributions in this method can be addressed as follows.

- Introduction of the definition for the individual expressions as the mapping in the deformation function. This definition achieve to extract the motion specific to the character quantitatively enables the methods to be used in various expression as input.
- Introduced of the mesh clustering algorithm that considers the base geometry of the model and the motion feature of facial expressions within the training facial expressions. This method allows the system to be able to be applicable even if the number of datasets are small.
- Introduction of a novel blending function of the deformation that avoids deformations unnaturalness with can be observed using a naïve linear blending. This blending method allows the system to be able to create plausible result even if the expression extremely differs.

3.2 Related Works

Highly realistic facial animation using blendshape animation is a one of the most used technologies and has been used many research related to facial animation.

Alexander et al., 2009. Facial retargeting technique which use blendshape animation is applied to numerous facial animationsBergeron and Lachapelle, 1985 Chuang and Bregler, 2004. Accordingly, several methods focusing on the facial expressions capturing and blending coefficients estimation finely has been well-studied. The prime focus of the some research is on tracking facial feature points in order to acquire the facial expressions from 2D video frames of a web camera Cao et al., 2013. Combining a depth sensor and video frames on the facial feature points, high quality facial tracking can be acquired to be used in retargeting Li et al., 2013 Bouaziz, Wang, and Pauly, 2013. These methods' approaches achieved high quality animation of the actor by aligning the generic face geometry onto the tracked facial data. However, since the facial expressions of the user are not identical to target character's geometry, small details of the face cannot be reflected. To add such details onto the the geometry, modelling high-quality blendshapes is required before the process in which the user fits its expression semantics to the human actor. Other works focuses on modelling and modification of blendshape. Some methods generates facial models from a single photos Pighin et al., 2006 or 3D facial scan data Zhang et al., 2014 Weise, Leibe, and Van Gool, 2007 where real-world geometry such as real human is necessary. Alternative methods creating PCA linear model has been studied while it requires numerous training data in order to create basis expression models using PCABlanz and Vetter, 1999 Ragnhild et al., 2003, Vlasic et al., 2005. Deformation Transfer Sumner and Popović, 2004 is a method in which the deformation of the facial expressions can be transferred from other characters; this approach has been a cornerstone method for creating blendshapes. While this method is applicable almost fully automatic, it does not take the geometrical feature or the difference in facial expression of the target model. Accordingly, small details that features the individuality of expressions cannot be transferred with this approach. Recent studies specified in certain parts of the face in which the methods improves artifact that have been yielded on the process of transferring the deformation of one character to another via Deformation Transfer Saito, 2013. While this method is capable of removing the artifacts by introducing virtual triangles inside the eyelid and lips, the character specific deformation cannot be removed because it only controls the parts where the virtual triangles exists. Some facial parts that have difficulty adding such

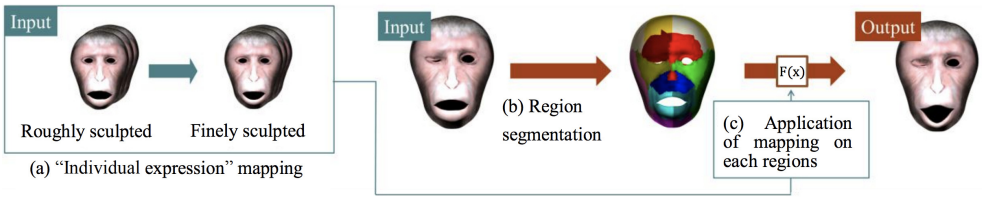


FIGURE 3.2: Workflow of our method.

virtual triangles will not be able to refine. Modification of the existing blendshapes using examples expression data of target characters to create character specific expression models is another popular method. Choe and Ko, 2005]Zhang et al., 2008 has proposed methods that can only be able to apply for generic model using captured sparse motion data with heavy restrictions. A method that is known to be effective for is the method proposed by Li, Weise, and Pauly, 2010. The blendshape will be modified to in order for them to be able to reproduce the training expressions by the combination of the resulting blendshapes. Although, the modification made by using this method is applicable for blendshapes that has significantly similar to the training examples, while in many cases the training examples are dissimilar. In such case, the users are required to prepare training examples of various kinds in order to appropriately modify arbitrary blendshapes. To summarize, there are three primary goals to be solved:

1) Limiting the number of the training examples 2) Use roughly created blendshapes as an input and propose method to modify them 3) Allows the method use arbitrary training expressions for the artists to be able to freely create the training data.

By achieving the goals above, the our method is applicable for any input training data and accessible for many artists to be used without the technical knowledge.

3.3 Proposed Method

3.3.1 Overview

As being addressed in Figure 3.3, we will use rigs created by using Deformation Transfer and training examples will be the input for Character Transfer. Our system initially generates the mappings from one deformations to another for every training

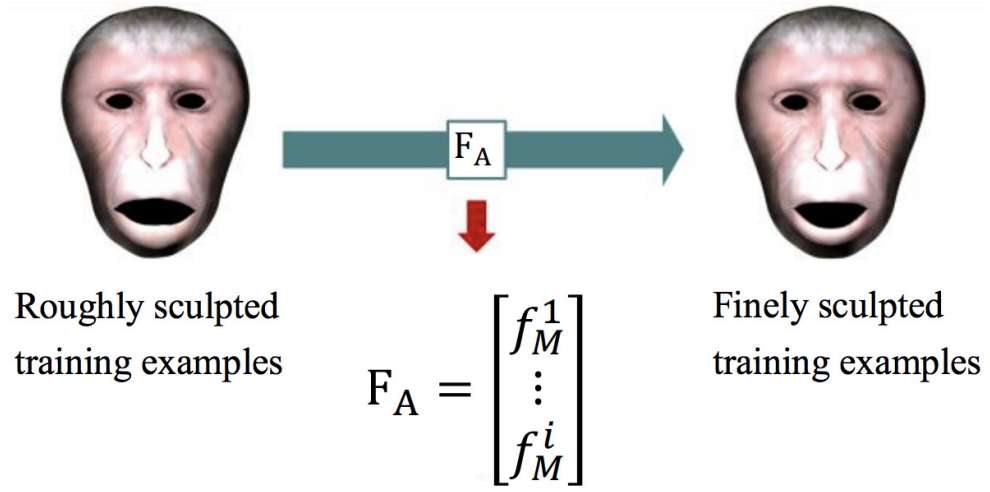


FIGURE 3.3: Creation of Individual expression mapping.

data and the deformation transfer blendshape using deformation gradient. Our system defines these mappings as individual expressions mappings and used to modify the expression generated by the Deformation Transfer. For the better representation, region segmentation will be applied. Character Transfer creates sufficient mapping that modifies the segmented region are apply depending the blending weight to the rough input blendshapes created by Deformation Transfer. Apart from the training data preparation, the method is fully automatic and applicable to any facial models with arbitrary expressions.

3.3.2 "Individuality expression" mapping

We have defined individual expression mapping from training expression model data in which they define the individual expressions. From the blendshapes expression model, for instance the model created using Deformation Transfer, we defined individual expressions by comparing expressions with the blendshapes created by artists. By defining the individual expressions feature as the deformation difference of the poorly expressive blendshapes and the finely expressive blendshapes.

To mathematically express these differences, we applied a deformation gradient to represent the deformation of the triangles. The deformation gradient for a single triangle is a 3×3 matrix in which it describes the rotation and scaling of the deformation required to translate from a non-deformed state to a deformed state.

The calculation of the deformation gradient is done by installing auxiliary vertex v_{j4}, which is computed by solving following equations:

$$v_{j4} = \frac{(v_{j2} - v_{j1}) \times (v_{j3} - v_{j1})}{\sqrt{(v_{j2} - v_{j1}) \times (v_{j3} - v_{j1})}} \tag{3.1}$$

Here, v_{jk} = k = 1, 2, 3 represent the vertices of the j^{th} triangle of the mesh model. Deformation gradient J_j is then computed by solving the linear system

$$J_j = V'_j V_j^{-1} \tag{3.2}$$

Here, V_j and V'_j are $3 \times 3 matrices that contain nonde formed and dde formeded gevectors of the$ jth triangle, respectively; That is,

$$V_j = [v_{j2} - v_{j1}, v_{j3} - v_{j1}, v_{j4} - v_{j1}] \tag{3.3}$$

$$V'_j = [v'_{j2} - v'_{j1}, v'_{j3} - v'_{j1}, v'_{j4} - v'_{j1}] \tag{3.4}$$

In our proposed method, we compute deformation gradients of finely as well as roughly created blendshapes from facial models with rest pose. More specifically, let be the facial expression of the training example, and s and t be the roughly and the finely created training expression data model, respectively. For the j^{th} triangle mesh, we compute deformation gradients of the rest pose models from s_m to t_m to form, $s^i_m \in R^{3\times3}$ and $t^i_m \in R^{3\times3}$, respectively, as

$$s^i_m = V'^i_{s\,m} V'^{-1}_{Ni} \tag{3.5}$$

$$t^i_m = V'^i_{t\,m} V'^{-1}_{Ni} \tag{3.6}$$

$$V'_{s^i_m} = [v_{s^i_{m2}} - v_{s^i_{m1}}, v_{s^i_{m3}} - v_{s^i_{m1}}, v_{s^i_{m4}} - v_{s^i_{m1}}] \tag{3.7}$$

$$V'_{t^i_m} = [v_{s^i_{m2}} - v_{s^i_{m1}}, v_{s^i_{m3}} - v_{s^i_{m1}}, v_{s^i_{m4}} - v_{s^i_{m1}}] \tag{3.8}$$

$$V'_{Ni} = [v_{s^i_{m2}} - v_{s^i_{m1}}, v_{s^i_{m3}} - v_{s^i_{m1}}, v_{s^i_{m4}} - v_{s^i_{m1}}] \tag{3.9}$$

Here, $v_{s^i_{mk}}\{k = 1,2,3,4\}$ are the vertices of the roughly created blendshape of the i^{th} triangle, $v_{t^i_{mk}}\{k = 1,2,3,4\}$ and are the vertices of the finely created blendshape of the i^{th} triangle. We then create a mapping in which it transforms $s_{i^m_k}$ into $t_{i^m_k}\{k = 1,2,3,4\}$. For each i^{th} triangle of s^i_m and t^i_m , we incorporate mapping $f^i_m \in R^{3\times3}$, which combines s^i_m and t^i_m as follows:

$$V^i_m = t^i_m {s^i_m}^{-1} \tag{3.10}$$

The definition of this individual mapping will be computed for every triangle of each training expression data. Therefore, for every set of training examples defines a mapping that enables to edit a given blendshape in a similar manner to that of the training data expression. Note that using specific deformation transfer technique is mandatory for this mapping tocreate the roughly sculpted input blendshapes of given expressions: any deformation transfer technique will be able to work as long as the training and test data is created with the same manner.

3.3.3 Hierarchical region segmentation

The mappings extracted from the training facial expression models are applicable to any expressions models of the given training examples. Although, since the facial parts are known to move independent to each other due to the muscle position, the mapping that matches the test facial expression can be completely different depending on how the facial part moves.

There are various mesh segmentation method exists to be used for facial expression transfer. Automatic segmentation methods that is used to segment the facial models does not considered the geometry feature of the model, for instance, the method proposed by Joshi et al., 2006. Another method is to use motion capture marker motion data as a input in order to segment the facial parts into several regions with similar motion feature Tena, Torre, and Matthews, 2011. It is difficult to apply this method since preferable our system is should not require facial expressions data of real human, in addition to the fact that this method does not take the

geometry of the facial model in consideration. Therefore, we propose a novel automatic region segmentation method in which it can be embedded effectively for our system.

The first step is to segment the facial model into triangular unit regions with the consideration of the geometry of the model with rest pose and the expressions of the training examples using a hierarchical algorithm. Multidimensional vector $P_i \in R^{D^1}$ D = 3 × 3 × (1 + M) is defined for each i^{th} triangle in the target shape as follows.

$$p_i = (v_1, v_2, v_3, d_{11}, d_{12}, d_{13}, \cdots, d_{m1}, d_{m2}, d_{m3},)^T \tag{3.11}$$

Here, M is the number of training expression data, $v_i \in R^{3\times 1}$ is a 3D positional data which contains the (x; y; z) spatial coordinates of the f^{th} f = 1,2,3 vertex the i^{th} triangle of the facial model with rest pose, and $d_m f \in R^{3\times 1}$ is the displacement vector which generated from V_f to the position of the f^{th} vertex in the i^{th} triangle of the $m^{th}\{m = 1, 2, \ldots, M\}$ training expression data. In order to execute the segmentation of the target facial shape effectively, recursive splitting of the polygons into two clusters will be applied. Multiplying p_i by a weight vector e_i , a center vector of cluster $c_t\{t = 1, 2\}$ can be computed by the energy minimization equation as follows:

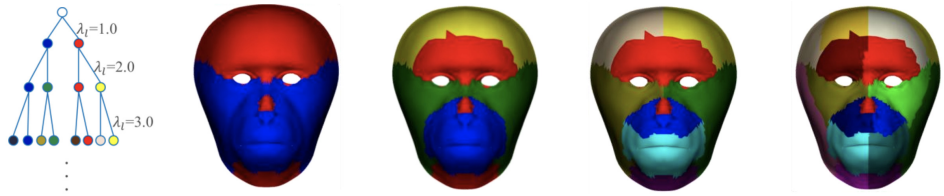
$$\phi_i = min_{cteC} \sum \|e_i p_i - c_t\| \tag{3.12}$$

$$P = p_1, \cdots, p_N \tag{3.13}$$

$$C = c_1, c_2 \tag{3.14}$$

$$p_i = (a_1^l, a_2^l, a_3^l, b_1^l, b_2^l, b_3^l, \cdots, b_{m1}^l, b_{m2}^l, d_{m3}^l,)^T \tag{3.15}$$

Here, l is the number of levels to recurs the process, and $_l$ is a constant weight parameter which is controlled independently by the number of recursive cluster. As shown in Figure 3.4, the value of $_l$ corresponds to a level of hierarchical clustering tree of the method. By editing $_l$ for each level of the tree recursively, the influence



(a) Value of λ_l in hierarchical clustering tree

(b) Segmentation at level 1 (λ_1 =1.0)

(c) Segmentation at level 2 (λ_1 =2.0)

(d) Segmentation at level 3 (λ_1 =3.0)

(e) Segmentation at final level (λ_1 =4.0)

FIGURE 3.4: Hierarchical region segmentation

level for the movements and geometry of the target can be modified as clustering progresses by formula (3.12). Figures 3.4(b), (c) and (d) represent the segmented regions in the level of each hierarchical clustering process. In higher levels of the clustering tree, the influence of geometry is set to be stronger and facial parts are segmented horizontally and vertically. This method can achieve the effective segmentation of the facial model for symmetrical feature in facial model such as the regions of the right, left, upper and lower of the eyelids. The method is also effective for the arbitrary facial models. In this paper, we have segmented the face model into 16 regions by our method as shown in Figure 3.4 (e).

3.3.4 The mapping blending

Its is possible to apply a single mapping extracted from a training example, while the modification the expression will be executed in a limited way. In order to make Character Transfer to be applicable for arbitrary expressions from the limited number of training facial expressions, creating a new mapping by blending mapping acquired by the training expression is required. To achieve this goal, we introduce a method to create mappings for each segmented region of the blendshape by blending the mappings. By blending the mappings in which the training expression data are similar geometrically to the input facial expression with estimated blending weights, we have successfully be able to generate a mapping that fits properly for the facial expression used in test set.

To quantify the similarity, estimation of the blending coefficients is achieved by estimating the naïve blendshape coefficients for each region of the segmentaion. For

the *rth* region of the model, blending coefficients are calculated by solving the following linear system:

$$V_r = B_r, w_r \tag{3.16}$$

Here, $v_r \in R^{3n\times 1}$ is the vector that represent coordinate values of vertices on the input blendshape and $B_r \in R^{3n\times M}$ is the matrix that contains coordinate values of vertices on the M training expression data. The quation above can be treated as a minimization problem below:

$$EW_r = \|v_r - B_r w_r\| \tag{3.17}$$

To be more specific, we compute the coefficients of the training data independently for each region by solving the minimization equation for each segmented region.

In the process of blending several training mappings, the straightforward approach to do this is to use linear blending of each element matrices: although, linear blending of the deformation gradient is highly possibile that the mesh triangles can be collapsed or fliped since the deformation gradient are capable of controlling both rotations and scales at the same time. In this system, novel blending method is applied to naturally blend several mappings by applying the interpolation method of two deformation gradient in which Kaji et al., 2012 proposed. The interpolation method proposed in this related work is achieved sufficient interpolation of two deformation gradients by using spherical linear interpolation of quaternion and exponential map of matrix. Let f_m^i be the m^{th} mapping, our system applies the polar decomposition Shoemake and Duff, 1992 to the mapping in order to decompose the matrix into rotation matrix $Rotf_M^i \in R^{3\times 3}$ and positive definite symmetric matrix $Symf_M^i \in R^{3\times 3}$. Next, we apply a alternative blending method to create the matrix according to blending coefficients effectively. For the *ith* triangle mesh, the rotation matrix is computed according to blending coefficients of the spherical linear interpolation in the quaternion space. The interpolated rotation matrix is computed using the degree of rotation according to the blending coefficient by solving the equation below:

$$Rot_r = \prod_{m=0}^{M} slerp(q_i, qRotf_m^i, w_{jm}) \tag{3.18}$$

Here, $slerp$ is an operator in which it performs spherical linear interpolation in the quaternion space, q_l and $qRotf_m^i$ is the quaternion of identity matrix and, $Rotf_m^i$ and w_{jm} is the blending coefficient for region r with which the i^{th} triangle is affiliated. As for a positive definite symmetrical matrix, our system applied a logarithm and an exponential map of the matrix. By applying this approach, the interpolation of positive definite symmetrical matrix according to the blending coefficient can be applied by solving the equation below:

$$Sym_i = \exp \sum_{m+0}^{M} w_{\ m}{}_{j} \log Symf_M^i \| v_r - B_r w_r \| \tag{3.19}$$

By using a blended mapping for both rotation matrix and positive definite symmetric matrix, our system solves the equation below to generate the mapping matrix:

$$Blendf^i = Rot_i \cdot Sym_i \tag{3.20}$$

This blending method offers the blending of the mapping with arbitrary blending coefficients while preserving geometrical property, for instance, the blended mapping does not unnaturally collapse or flip. The definition of the blended mapping modifies the deformation gradient from roughly created input blendshapes by applying this modification mapping with the ones of finely created blendshapes. Therefore,the blended mapping is applied to the ith deformation gradient which the facial model with rest pose and test facial expression is defined can be represented as follows:

$$B = [Blendf^1 \times J_1, \cdots, Blendf^i \times J_i]_T, \cdots, Blendf^N \times J_N]_T \tag{3.21}$$

In the final procedures, we solve for x, y and z in the following equation which coordinates for the vertices of the output blendshape created by Character Transfer, $X' + [x'_i, \cdots, x'_N]$ is as follows:

$$\min \|B - AX'\| \tag{3.22}$$

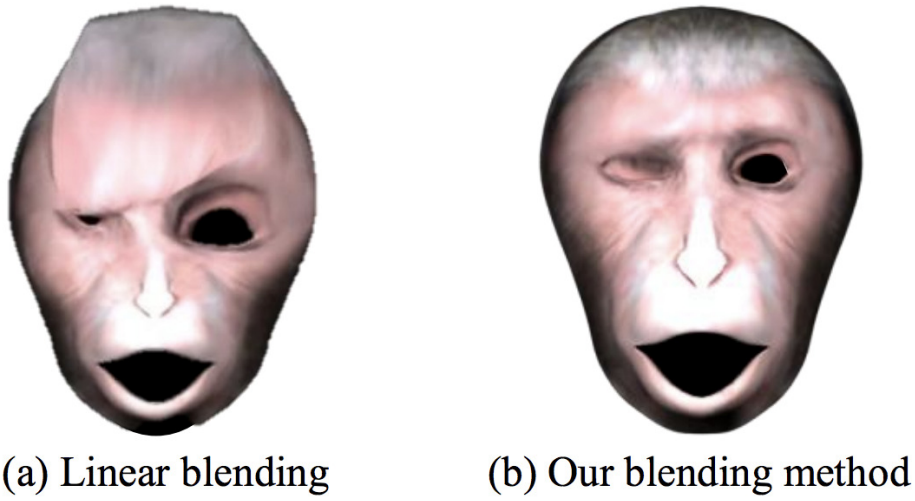


FIGURE 3.5: Illustrative comparison of linear blending and our blending method, our blending method successfully avoided unnatural deformations.

Here, $A \in R^{3M \times N}$ is a large sparse matrix in which $AX' \in R^{3M \times 3}$ is the deformation gradient defined between the facial model with rest pose and the blendshape generated by Deformation Transfer, in which it is method proposed by Sumner and Popović, 2004. Above equation can be formulated into the linear system shown below:

$$X' = (A^T A)^{-1} A^T B \tag{3.23}$$

By solving this method in order to directly compute the coordinate value of the vertices, the computed positions of the vertices preserves the continuities across the regions in which the system segemented. Since our method solves minimization problem in which the semantics of the minimization are significantly similar to the equation solved in Deformation Transfer, Character Transfer can be seemlessly incorporated into the framework of Deformation Transfer. The property offers the other methods, which is also inspired by the Deformation Transfer, to be implementable to the given equation, including the modification method proposed in Saito, 2013. Furthermore, since Character Transfer is applicable via only few training examples and roughly created input blendshapes, Character Transfer support the modification of the blendshape prior to other modification methods which allows more flexibility in our method.

3.4 Results

Figure 3.7 shows that the blendshapes created using Character Transfer in comparison with blendshapes created by an artist and using Deformation Transfer. We have tested on the blendshapes of a monkey with only the front of the face, in which the consisting vertices are 5K. We asked an professional artist to create four training expression data to be the input for character transfer. As for roughly sculpeted input blendshape of arbitrary expressions, we used Deformation Transfer by transferring the expressions created for of human face model. The triangle-wise correspondence between monkey model and human model have been created by adopting the semi-automatic correspondence computation method used in Sumner and Popović, 2004. The computational time for all the procedures to create one blendshape was approximately 13 seconds using Intel Core^{TM} i7-2600 CPU without parallelization nor computation using GPUs. Character Transfer required no manual parameter settings except for the correspondence which was addressed above. However, the blending coefficient can be set manually if the user prefers to have the control over the deformation. We also created facial animations using the blendshapes created via Character Transfer blednshapes and compared the result to the one created using the Deformation Transfer blendshapes. The facial animation videos consist of random facial movements created from 14 blendshapes for 270 frames. The animation using the blendshapes created by our methods are considered more similar to those sculpted by the artist compared with the result created by Deformation Transfer result.

The significant cause of the artifacts yielded for the result using Deformation Transfer is the differences of geometrical consideration of the rest pose. Such artifacts are visible around lips in Figure 3.7 because of the size difference of target models lips and those of human models. As a result, the unnatural deformation around the lips have been observed; however, since our training expression data includes the geometrical difference to be in consideration on their lips, the artifacts can be removed successfully. The artifacts around mouth and eyelid have been modified in the comparison of the blendshape created using naïve Deformation Transfer. While these artifacts can be modified using alternative modification method proposed in

Saito, 2013, such work is limitated in terms of the use case that refining arbitrary movement of opening and closing eyelids could not be defined effectively when the target model had a significantly large topology deference than that of the source model. In the case of Character Transfer, such limitation will not be the problem as it is possible to create deformation of an arbitrary amount of opening and closing of the eyelids by only creating training expression data with the movements of eyelids. From the these results , the goals we set out in the previous section has successfully achieved which indicates that Character Transfer has versatility and applicability to many animation creation scheme that require facial animation creation procedures.

3.5 Evaluation

The fundamental goal of our method was to generate blendshapes in which they are geometrically similar to those created by an artist. To show the effectiveness of our approach, we have evaluated the similarity geometrically between the two by the vertices distance between of the blendshape generated by Character Transfer and those generated by Deformation Transfer. We show our results using an error map in which error is shown by the color of the vertices in Figure 3.7. The maximum error is shown in red, whereas the minimum error shown in blue. Errors involving the eyes and mouth can be observed as significant difference for the blendshapes created by Deformation Transfer. The average root means square error is calculated for all vertices of three blendshapes generated by Character Transfer and those generated by Deformation Transfer . From the results are summarized in Table 3.1, our method is proven to be effective in that the blendshapes sculpted by Character Transfer has lesser error on average. We also computed root means square error for all vertices of the blendshapes using our hierarchal region segmentation approach more specifically; we subjectively defined segmented regions for three blendshapes, which are shown in Figure 3.6. Results are summarized in Table 3.6 which reveals that our region segmentation algorithm effective comparison to the segmentation initiated by the users. We also evaluated the root means square error on all vertices of the facial animations created by using sculpted by Deformation Transfer. The comparison results are summarized in Table 3.2 and it shows that the facial animations created

TABLE 3.1: Comparison by the result of root means square error on all vertices when creating facial animation.

Sumner et al. 2004	Region defined subjectively	Character Transfer
3.629 cm	0.764 cm	0.611 cm

TABLE 3.2: Comparison by the result of root means square error on all vertices when creating facial animation.

Sumner et al. 2004	Character Transfer
1.032 cm	0.726

by our blendshapes have significantly less error as compared with those that used blendshapes generated by Deformation Transfer. One goal in which we have set out to was to make Character Transfer effective to blendshapes with arbitrary training facial data. We applied Character Transfer to the facial expression which is dissimilar from the training example. Although results showed that Character Transfer is not able to create an exactly similar geometry, some of the features were successfully modified the way in which the lips deformed. Modification for extreme large facial expressions can be difficult to apply using modification yielded of Li, Weise, and Pauly, 2010 because this method is only applicable when the training expression data has similarity with the input facial expression. To address the limitation of method, we observed that Character Transfer is capable of the modification of facial expressions in which it is impossible to estimate the blending coefficients given the set of training examples. To make the lesser artifacts, facial expressions on training expression models are expected to be as extreme as possible to create large character individuality space to be covered. This disadvantage also has affect for the hierarchical region segmentation algorithm of Character Transfer. Due to the fact that our hierarchical region segmentation method considering the way in which training expression models are move their facial parts in lower parts of tree, the segmentation results will not always be effective; however, it is an improvement over the modification method by Li, Weise, and Pauly, 2010 because blended facial expressions of training expression model is tha result to be similar, is not exact.

FIGURE 3.6: Subjectively defined Regions

3.6 Conclusion and Future Works

In this chapter, we presented a system called Character Transfer that modifies roughly created input blendshapes of arbitrary expressions to sculpt individual expressions. This example-based modification method to add individuality can be applied in arbitrary expressions, not limited on geometrical restriction nor geometrical feature of the rest pose, by blending the modification mapping extracted from training expression models. Using our blending method, we have successfully avoided the visual artifacts often introduced by the blending mapping. In addition, the number of training examples can be reduced using blending several mappings to generate a new mapping. Character Transfer can automatically generates the blendshapes using very few numbers of training expression data. To extract the modification feature using the idea of blending them. we also introduced a method to generated the segmentation of the mesh taking the geometry and expressions of examples into account. The significant contribution in this method in this chapter is that we have formulated a novel method of modifying blendshapes that can be adopted even when the number of training expression model is reduced. To the best of our knowledge, such property of the result has yet to be achieved in previous research; furthermore, the novel segmentation and mapping blending approach we have proposed.

As a future work, we aim to generate a system that are capable of selecting effective training expression data. The training data are currently selected subjectively; by using a system that is able to systematically suggest the training examples, the versatility of our method significantly increases because of the method being fully automatic. We are also interested in applying an improved approach to estimating more effective blending coefficients for Character Transfer. To date, only the naïve estimation of blending coefficients for each region has been investigated, but Character Transfer yields the artifacts due to the fact that the coefficients was not solvable in the effective range.

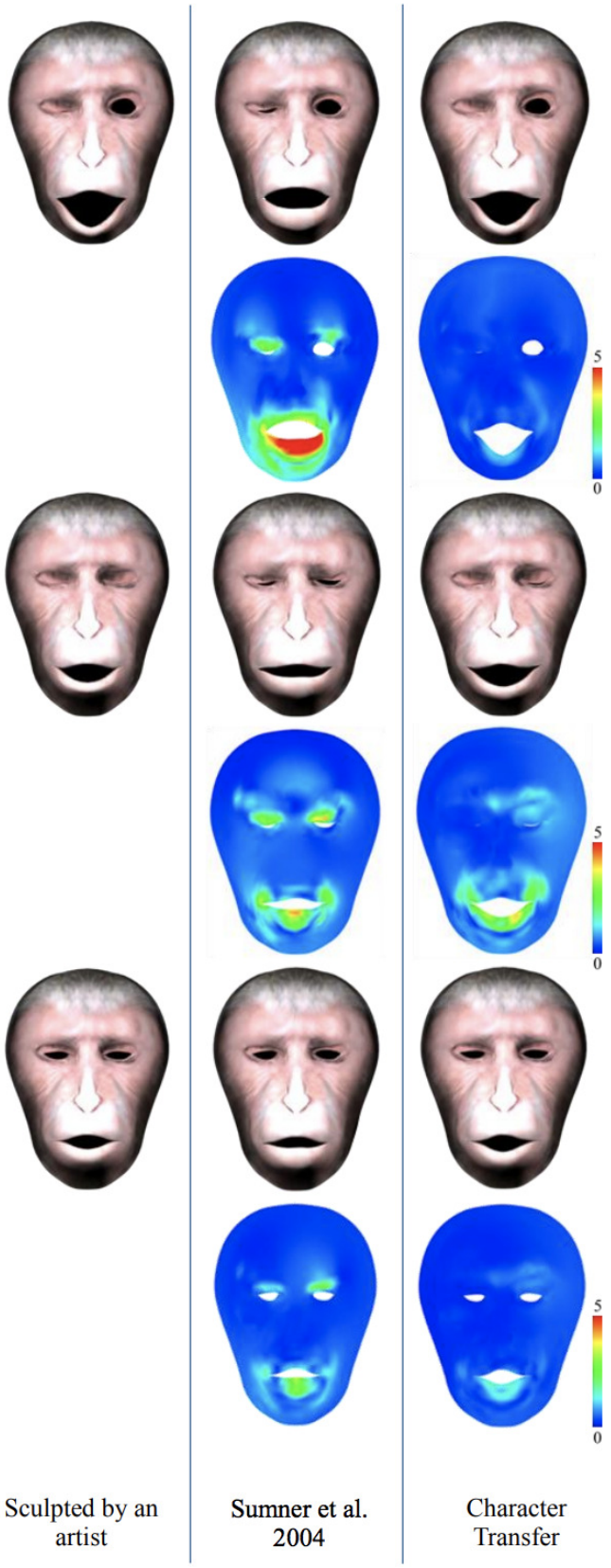


FIGURE 3.7: Comparison with the grand truth, previous method and proposed method. The result indicates that our method successfully modifies the result created by the previous methods.

Chapter 4

Singing Facial Animation Synthesis using Musical Information and Singing Voice

In 3DCG speech animation, head rotations and facial features other than mouth motions are considered to be equally important as mouth motions. Creating such motions in singing animation requires a huge amount of work by hand. Despite its demand, none of the research has ever been conducted to generate realistic singing animation. In this work, we present singing animation synthesis method with the information associated with the song. Using singing voice, song information and the mouth motion as input, the proposed method are able to create realistic head motion and expressions around the eye fully automatic. These input data is easy to create only from the singing voice and song information. We have conducted the numerical evaluation as well as subjective evaluation on the naturalness and the synchronization to the singing voice and song to show the effectiveness of our method.

4.1 Introduction

In 3DCG Character Animation creation, facial expressions plays tremendously large role when it comes to the impression of the character. Speech animation, the animation that includes speech sequence to the animation, are created in two layers in production. First, mouth motions are created in order to match the contents of the speech by keyframe by keyframe. After the mouth motion are set, the mouth motions set by keyframes are interpolated over the frames and the keyframe animations are created. After the 1st layer of the process, the facial animation of the eyes and forehead and head rotations are applied keyframe by keyframe to match the speech animation. The reason that this process is regarded important is that the eyes and

forehead animation and head rotations have strong correlation with the content of the speech and will not be determined by it's own. The mouth motion animation are predominantly created by the script that the director or screenwriters create, and will not largely change depending on who the artist creating the character is, what kind of individuality the character have or how the characters will act depending on the scene. Although, the synchronization of the mouth motion are considered to be the essential part of the high quality speech animations and requires huge amount of the artist works. The eyes, forehead animation and head rotations, on the other hand, have very large role on how the characters act each of the scene. Not only the emotion, but also habit of the expressions, how the artist and directors want the characters to act will take huge role on creating these motions and requires huge amount of burden to create such animation. From such background, the technique to creating the speech animation are considered to be one of the most difficult animation contents in 3DCG animation despite of it's high demand, and the technique to create speech animation from the accessible inputs are demanded by industry.

One of the examples in speech animation is a singing animation. Singing animations are highly demanded contents in industry with its popularity in musical films and Anime. The facial motion in singing animations have some common features with speech animations but mostly a lot different from the speech animation and known to be difficult to create. For example, the general speech animation such as character just speaking a basic sentences or having a speech, the mouth motions are required to sync the script of the speech while the other facial features and head rotations does not change largely by what the characters speak about. Accordingly, many methods that are proposed by by the researchers to create speech animation automatically, the facial features of outside of the mouth are mostly reused from the sequences of the other speech animation and it would seems as natural as it was originally created. On the other hand, the mouth motion of the singing animations will not be as different from the how the other speech animation creates them, but the other motions will change differently according to many circumstances. The way the singers sing, the volume of the singing or the rhythm and the beat of the song, the animations changes each by each. In addition, such animation are individual to each other and requires not to be reused from the other character even the song

they sing is identical. Accordingly, the facial animation for the singing animation requires to be very different from what other speech animation creates when it comes creating the animation outside of the mouth and head rotations.

From such background, this chapter proposes a method to automatically generate facial expression animation at singing from few accessible inputs, which is singing voice information and song information. In this method, in addition to the singing voice information used in the speech animation generation method, we use song information to estimate expression parameter at singing and head rotation information in order to create a facial expression animation specialized at the time of singing. Paying attention to the fact that the behavior at the time of singing has a high dependence on the song, in addition to the volume and the acoustic feature amount at the time of singing highly correlated with the head rotation information at the time of singing, information on the rhythm and melody are adopted. Furthermore, mouth shape information which changes according to lyrics is estimated from singing voice and adopted as input data. By making learning of these data into a deep learning model which take time series into consideration, the method made it possible to generate facial expression animation at singing which was difficult only with singing voice information. In this paper, in addition to quantitative evaluation of these results with measurement data, we verify the naturalness of the generation result and whether animation unique to each singer can be learned by subjective evaluation experiment. By evaluating this, we will discuss the validity of the input parameters and the stability of the generation result.

Contributions of our method in this chapter are as follows:

- **Achieved singing animation synthesis using song information which was known to be difficult from other speech animation synthesis methods**
- **A more efficient learning result was realized by performing data compression suitable for patterns frequently appearing in facial expression animation at singing.**
- **By combining high dependency parameters such as beat information and pitch information with parameters that depend on individuality such as voice, we succeeded in generating more accurate singing animation.**

4.2 Related works

Facial expression animation creation technology has a wide range of studies including face shape modeling **ao2017sparse** facial expression animation generation Cao et al., 2017, texture generation Huynh et al., 2018, speech animation generation Saito et al., 2017 There. In recent years, research has been proposed to automatically generate various elements related to the face using simple data that can be obtained from mobile terminals as input as well. Against this background, there is development of research using deep learning. A learning model called convolution neural net is often used for face modeling, facial texture generation technology Saito et al., 2017, facial image generation technology Selim, Elgharib, and Doyle, 2016, and the like because of its high performance.

4.2.1 Facial animation synthesis using Deep Learning

For generation of facial expression animation, effectiveness of the deep learning method which can learn time series data called recursive type neural network (hereinafter referred to as RNN) has been shown. Lu et al., 2017 RNN is a deep learning model that can input continuous information used for natural language processing and other more, it is often used for continuous learning data with respect to time series. Among them, the deep learning model called LSTM Hochreiter and Schmidhuber, 1997 shows high learning accuracy in many studies and is used in various fields.

The feature that continuous data can be learned with respect to the time series is highly demanded in research field to estimate expressions and mouth shapes. Accordingly, various methods using LSTM have been proposed in 3DCG Kim et al., 2018 proposed in recent years especially for automatic generation of speech animationSuwajanakorn, Seitz, and Kemelmacher-Shlizerman, 2017. In speech animation generation method Zhou et al., 2018 which estimates the mouth shape of 3DCG character from speech, speech information of volume feature and audio feature are applied to be used as input to estimate mouth shape and tongue movement. In this research, the method is only specialized in estimating the mouth shape of speech animation, and no consideration has been done to generate animation other than

mouth. In the method of generating a speech video from speech Kim et al., 2018, like the previous method, it estimates the mouth shape by using the volume and the music feature as the input of the LSTM and generates the frame-by-frame time series image of the expression which matches the mouth shape. While this method can generate high-quality speech video the appropriate facial expressions other than mouth are selected from a vast data set and reused to fit the mouth motion. Therefore, learning requires a large amount of data set, and it also indicates that this method has not been able to generate facial expression animation.

4.2.2 Animation Synthesis from Music

Studies focuses on the movement of characters in accordance with music have also been proposed. Especially, in dance motion generation, in which the dance motions are strongly correlated with music, various methods have been proposed using the correlation on music characteristics and movement. Kakitsuka et al., 2017 Fan, Xu, and Geng, 2012 Other research have been proposed to make motion generation of characters more accurately by using features unique to music such as music features, rhythms, pitches, and song structure. Fukayama and Goto, 2014 Intending from this method, we focused on characteristics peculiar to music and adopted it as an input parameter at learning process.

4.3 Relationship of facial animation and singing voice and song

Various studies have been done on facial expressions and head movements during singing, and it is known that there is a strong correlation. Thompson, Graham, and Russo, 2005 In this chapter, we will describe the relationship between facial expressions at the time of singing and movements of the head and singing voices and music, which support the method after the next chapter.

It is known that there is a correlation between the pitch and the movement of the head with respect to expression and head movement at the time of singing. In the field of psychology, experiments that tell the pitch from the appearance at the time of singing and the movement of the head have been reported. Quinto et al., 2014

By investigating the movement captured using motion capture attached on the face and the pitch, it has been reported that human can perceive the pitch only by looking at the facial movements. Moreover, it has been reported that there is a correlation between the pitch of the head and the movement of the head, even in experiments in which the height of the sound is indicated from the video of the singer, even with the portion of the head is painted in black. Thompson, Russo, and Livingstone, 2010 From these experiments, it can be said that there is a movement specific to singing for the head during singing, and there is strong correlation with the characteristics of singing voice. Furthermore, considering the perspective of the viewer on the singers, it is understood that the viewers singing and the movement are observed have strong relationship. From these facts, it is possible to confirm the importance of reproducing expressions matching the singing voice information and the movement of the head, and it is adequate to estimate expressions and head movements from singing voice information it is conceivable that.

On the other hand, from the study of Thompson, Graham, and Russo, 2005, it can be seen that there are some problem in estimating facial expressions from only singing voice and head rotation information. When considering estimation of singing animation only from singing voice, it is understandable that movement can be estimated for the frames in which singing voice exists, but there are parts where singers do not utter within many songs. At the frames that the singers do not utter, many singers move in accordance with the rhythm, preparations before and after the utterance etc., they perform singing peculiar movements even without utter. Although, if only singing voice is used as an input, since there is little information in certain frame, the expectation is that estimation of appropriate motion becomes significantly difficult. Moreover, from the experiment of Quinto et al., 2014, it is known that the strong correlation with singing voice is head rotation information. On the other hand, it is also possible to estimate the pitch even if the face can not be seen, and there is possibility that the singing voice is not sufficient information. If that is the case, it is required to understand what exactly would affect the head rotation and the facial expressions when singing. As described above, with input of only singing voice, it is conceivable that there is a problem in estimation of motion that depends only on musical piece information such as silent period.

Therefore, in this research, we consider adding song features that are considered useful for estimating motion dependent on songs, and mouth shape information that is considered useful for estimating detailed expression estimation as input. Musical piece information has characteristics such as beat and music score information, which are also present in parts that the singer do not sing. Such song features can express features that lead to motion estimation, such as movement to take rhythm in accordance with beat at a place where singing voice does not exist, how far singer deviates from original music notation Conceivable. Additionally, one of the facial expression features that is considered difficult to estimate by singing voice alone is that expressions created by interaction of multiple facial muscles. With it's complex relationship, some facial muscle enhance the motion of the others to move. Therefore, with the consideration that it is possible to estimate expressive parameters with strong correlation with such features, there is high possibility that facial expressions can be estimated only from singing voice information even the voice itself only exists partially. Many of the research on facial expression synthesis focuses on this feature and shows its effectiveness, such as suggesting a method of compressing dimensionality of facial expressions. Therefore, I focused on the mouth shape information estimated from the speech which achieved success with speech animation etc. The mouth shape is also a part of important facial expressions and it can be considered that it is possible to estimate how the other facial muscle acts from features such as change intensity of the mouth shape and how to open it. In this way, by adding music information and mouth shape information, we made a hypothesis that information on music can be interpolated the part that misses the singing voice, and add more high dimensional information to the learning process. In this research, in order to demonstrate this hypothesis, we examine how these information can affect the estimated result by comparing the results of learning these pieces of information to the same deep learning framework.

4.4 Proposed Methods

In this chapter, we will describe in detail the method of generating facial expression and head rotation information other than mouth at singing. Based on the hypotheses

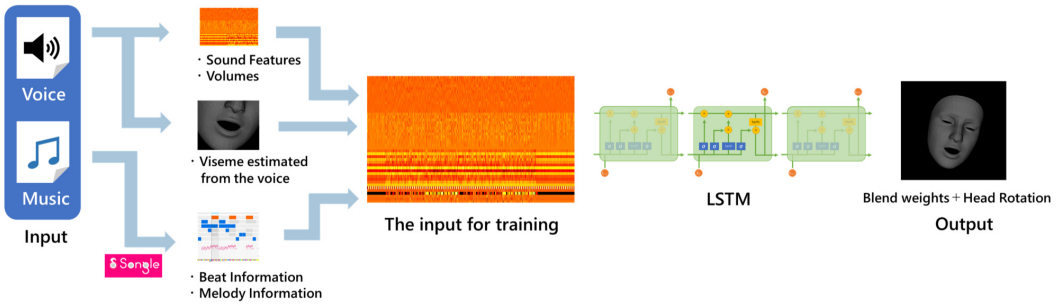


FIGURE 4.1: Workflow of the proposed method

in the previous chapter, we describe singing voice information, music information, and mouth shape information used as inputs. Next, we will describe how to obtain head rotation information and expressions other than mouth as output targets. Finally, as an expression estimation method using these, we will describe the estimation method for facial expression and head rotation information using LSTM which is a machine learning algorithm suitable for time series data. A schematic diagram of the proposed method is shown in Fig 4.1.

4.4.1 Dataset Creation

With the improvement of the facial expression estimation technique in recent years, it has become relatively easy to estimate facial expressions from video and head rotation information. In this research, we propose a facial expression generation method using machine learning, using learning data estimated from the video of singer singing in front of the camera. Here, we consider the construction of the data set, which is important when using the machine learning generation method. In order to improve learning efficiency of machine learning, it is necessary to convert input data and output data to data with high correlation with output. In this section, we describe a method of converting singing voice information, music information, and mouth shape information, which are input data, into more meaningful feature quantities. We will also describe the data set construction of facial expression data and head rotation information as output.

4.4.2 Singing Animation Information

First, input information obtained from singing voice will be described. Besides the lyrics, the singing voice includes the difference in singing characteristics of singers. Specifically, it includes acoustic features such as volume and utterance. Based on the psychological experiments in the previous section, it is considered that such information is useful information for estimating head rotation information. Therefore, in this research, it follows the feature quantity used in the speech animation generation method of Zhou et al., 2018. The volume and the acoustic feature quantities focusing on the volume and the acoustic features used when estimating the mouth shape of the speech animation are applied. As the sound volume, the power which is the time average value of the square of the waveform amplitude is applied. Mel frequency cepstral coefficient (hereinafter referred to as MFCC) Logan, 2000 widely used for speech recognition and timbre analysis in music information processing field is applied as acoustic feature quantity. In addition, in order to provide information on the temporal change of each feature, many research applies the sound volume and the MFCC as well as it's first-order differentiated and second-order differentiated values. In this study, we also adopted the volume, MFCC, and the first-order differential in the time direction, the second-order differentiation as the feature of singing voice.

4.4.3 Song Information

Music has high-dimensional information which is difficult to analyze only from singing voice information, but certainly has the information on how singers sing. Features such as characteristics at places where singers do not utter, features not dependent on differences in singers, and information specific to the songs of the songs, so that the characteristics of the songs themselves can be more strongly expressed. Such features are considered useful for estimating various information such as utterance timing and the motion induced by beat. In this research, we use the analysis result of the system proposed by Goto et al., 2011. By using this system, various music information can be obtained accurately and easily. Among the information that the system provides, we adopt melody and beat information in this method. Melody

is pitch information at the time of singing original songs. Similar features can be presumed from singing voice information, but there are times when singers use expressions such as "tame" which uttered at a timing slightly differ from the original beat depending on the singer. These pieces of information observe in time series, it differs little by little. In order to capture these expressive features, we added this information separately from the acoustic features of singing voice in this research. The beat information is the count of the beat defined for the song. Since the beat means the relative position in the measure, it is possible to express which timing of singing is the respective beat. By adopting the beat information, it can be considered that the change according to the beat of the singing way can be estimated. In addition, beats are also characterized in parts that are not uttering, so it is considered useful for estimation in singing parts that are not vocalized. For the beat information, a time series length vector in which one of each beat is included within that vector used as an input.

4.4.4 Mouth shape information

In this research, only singing voice information and music information are used as input. Therefore, one other feature we can use is the mouth shape information estimated by the method as the mouth shape information. Zhou et al., 2018 In this method, we estimate the basic visual features of speech that represent the position of the face and mouth when reading words called visemes. The feature of viseme is to treat the same role as phoneme in the voices, but as visually. This makes it possible to express the characteristics of how to move mouth which does not exist in the sound itself. When viseme is similar in how to move the mouth, since the mimetic muscles to use are the same, we can capture the characteristics of facial expression change with only singing voice information by using it as input. In the method of Zhou et al., 2018, it estimates 22 feature quantities that characterize Vizem. This feature quantity includes the strength characteristics of the viseme, the feature related to the connection between the visemes, and the correlation feature of the jaw and the lip, and the visual similarity is calculated. In the this method, these 22 features were measured on a frame basis and adopted as input information.

4.4.5 Facial Expression and Head rotation acquisition

The purpose of this research is to generate expression animation of expressions other than mouth at singing and head rotation information. There are various method to control the facial animation such as rig bases, bone basses, muscle bases. Among them, a technique called blendshape animation was adopted in numerous CG application for the facial expression model. Blendshape animation is an animation production method that deforms facial expressions by determining corresponding vertex positions by weighted linear sums of a plurality of different basic facial expressions that have the same mesh topology. Joshi et al., 2006 It is widely used in many researches and applications in recent years, and it is used as a generic but versatile facial expression model creation method. In this study, a blendshape model expressed by facial expressions with 51 basic expression models was used, and weighting coefficients of basis expressions in each frame were used as expression data. For estimating the weight of the basis expression in the learning data, the expression estimation method Li et al., 2013 using the 2D facial feature point and 3 dimensional face shape estimation is used. From the weighting coefficients for each frame of the basis expression obtained by applying this method to the animation sung by a person, the weight coefficients of the 19 kinds of base facial expressions, which are the basic facial expressions other than mouth. These parameters are adopted as output data. Also, when estimating the weight of the basis facial expression, the three-axis head rotation information measured at the same time is also adopted as the output data.

4.4.6 Input Data and Output Data

The singing voice information, the music information, the mouth shape information, the weight coefficient of the expression other than the mouth shape, and the head rotation information are different in frame rate and value range, respectively. Therefore, normalize the frame rate and value range, then input and output. For singing voice information, song information, and mouth shape information, time series data of the same length were made by unifying the sampling rate to match the expression

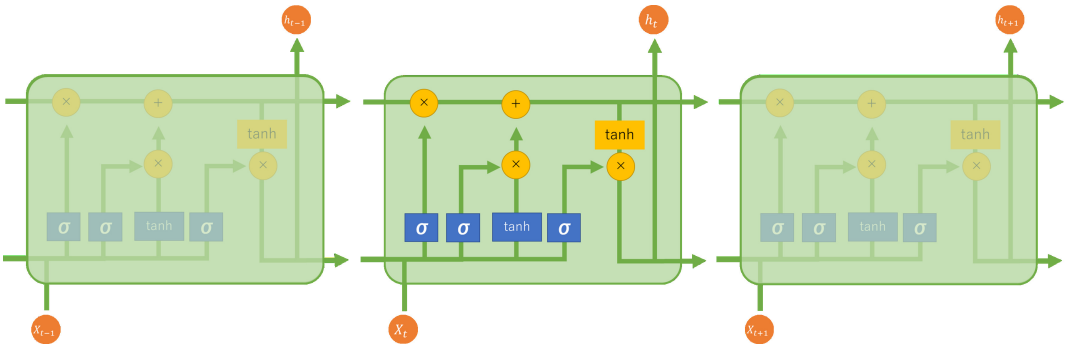


FIGURE 4.2: Abstracted Image of LSTM

data. In addition, each data was normalized to a numerical value from 0 to 1 according to the maximum value and the minimum value of individual data. As described above, the input data used is constructed as follows; 39 dimensions of acoustic information, 3 dimensions of volume information, 3 dimensions of song information, 22 dimensions of mouth shape information. For the output data, weighing coefficients of facial expressions other than mouth totaled 19 dimensions, and head rotation information was 3-dimensional data. Using the above data, we adopt the data of 67 dimensions is used as a input data, and 22 dimensions as the measurement time data are used as output data respectively.

4.4.7 LSTM Singing Animation Estimation

In this research, by using depth learning, expression parameters and head rotation information at the time of singing are estimated from input singing voices and music information. As described above, the input data and the output data respectively change in time series, and each has strong correlation in the time series direction. LSTM has been proposed as a deep learning method with high estimation performance for such input and output of time series data. In this section, I will outline the basics of LSTM and then describe the details of the network used in this research.

LSTM (Long-Short Term Memory)

LSTM is a type of RNN that is often used as learning of time series data. RNN is a model that learns time-series data by setting the output value of the middle layer of the neural network as input when learning the next data. On the other hand, in

general RNN, problems such as gradient elimination and gradient explosion occur, so learning was known to be difficult. LSTM Hochreiter and Schmidhuber, 1997 was developed in order to solve these problems. LSTM is a model that solves these problems with a mechanism that imitates human short-term memory and long-term memory. By adopting internal mechanisms such as input gate, output gate, and forgetting gate, it made it possible to learning while choosing necessary information from the input information, gradient loss hardly occurs in time series data, and learning with high accuracy. In recent years, it has been adopted in learning of various time series data, and it is known to result in high learning performance. In this study, LSTM network was constructed with reference to the LSTM network used by Suwajanakorn, Seitz, and Kemelmacher-Shlizerman, 2017 and others. Although this previous method use one layer of sigle-directional LSTM, the network used in this research is three-layer unidirectional LSTM because the input has high dimension. Adam function is used for optimization, hyperbolic tangent which is a hyperbolic function is used for the activation layer of the final layer, and the L2 norm with the correct data is used as a loss function. A schematic overview of the network is shown in Fig. 4.2.

Learning using Data Compression

In this research, facial expression parameters and head rotation information are used as output. Although it is possible to output such data as it is, it results in very low learning efficiency. This is because the dimensionality of the facial expression parameters to be output is significantly large, and it is a major factor that affects to the efficiency of learning. In order to solve this problem, in this research, efficient learning is performed by using dimension compression method focusing on facial expression singing and facial expression. Focusing on expressions at the time of singing, facial expressions change with various factors such as the height of the sound and the structure of the music, but it is known that facial expressions with high similarity and head rotation are performed in singing sections with high similarity. Livingstone, Thompson, and Russo, 2009 Also, it is also known that many songs have a repeating structure, and repeatedly similar expressions appear. Also it is known that there are many bilateral symmetries in facial expressions Kanade,

Tian, and Cohn, 2000 and has many features such that the same parts on the left and right move at the same timing. In this way, since there are many parameters that change depending on the similar trend in the output data, we apply dimensional compression for more efficient learning.

Therefore, in this work, the basis expression and the base motion are different according to the individual, and it is verified that the individuality is original for only them, even if the music changes. The base motion does not change and only the time series data of the weighted coefficient It is assumed to change. As described above, since facial expression changes during singing are caused by the height and volume of the sound, the base expression at that time does not depend on the music, and only its frequency of appearance changes. Therefore, baseline expression in individuals is estimated before learning, and data in which those expressions appear appears as time series data. Consider processing this time series data as output at learning.

Non-negative value matrix factorization (hereinafter referred to as NMF) is known as a dimension compression method capable of such expressions. NMF is a mass analysis method that can decompose non-negative data into additive basis components. In Lee et al., 2002, NMF is considered that the observation vector is represented as a weighted sum of a plurality of basis vectors for such an amount that additivity is satisfied, and a base vector that best explains the observed vector As a weighted coefficient. NMF is used for various studies as a dimension compression method of time series data of nonnegative values such as voice data and image data.

In this work, time series data of output facial expression parameters is regarded as a two-dimensional matrix, and NMF is applied and matrix decomposition is performed as basis vectors and weighted coefficients. As described above, the expression parameter can be expressed as a weighted linear sum of the base expression data, and since its expression data are all nonnegative values, the affinity with NMF is high. By using NMF, it can be represented as frequently appear base motion and time series data of the weighted coefficient. Let the character have N blendshapes, the facial expression data vector will be:

$$X = WH^{\mathrm{T}} \tag{4.1}$$

will be executed to create 2 Non-Negative Matrix, $W = [w_1, \ldots, w_K] \in \mathbb{R}_+^{M \times K}$ and $H = [h_1, \ldots, h_K] \in \mathbb{R}_+^{N \times K}$. Here $w_k \in \mathbb{R}_+^N$ and $h_k \in \mathbb{R}_+^t$ is the facial expression basis vector and the other is weight vector. In this case, H will be the output data of the proposed method.

The output of this learning method is time series data of weighting coefficients of NMF. Therefore, in order to actually output as an animation, it is necessary to reconvert to the original facial expression parameter. This is possible by multiplying the matrix representing the basis motion generated by the NMF by the time series data of the weighted coefficients outputted and reconverting it into the expression parameter. The expression parameter obtained as a result becomes the original 22-dimensional expression parameter.

As described above, by using the singing voice information, the music information and the mouth shape information as an input and use the weighting coefficients for the time series obtained by dimensionally compressing the weight coefficients of the base facial expressions other than the mouth by the NMF as an output using LSTM. In the case where the music information has been analyzed, in the test phase, singing voice information and mouth shape information are estimated from singing voice data and put into the learned LSTM model by creating expression animation at the time of new singing voice data.

4.5 Results

In this section, we will describe the facial expression animation result created by the proposed method. We describe the environment of the test used to create the results at the outset, and then show the animation result generated using the proposed method.

In this experiment, we used singing voice data and facial expression data recorded and recorded in the music studio by three singers whom they are Japanese women sang in Japanese. Singing voices and facial expressions when each singer sings two songs of popular music are recorded as an acoustic signal and a video. After that, expression and head rotation information were estimated for the captured using video

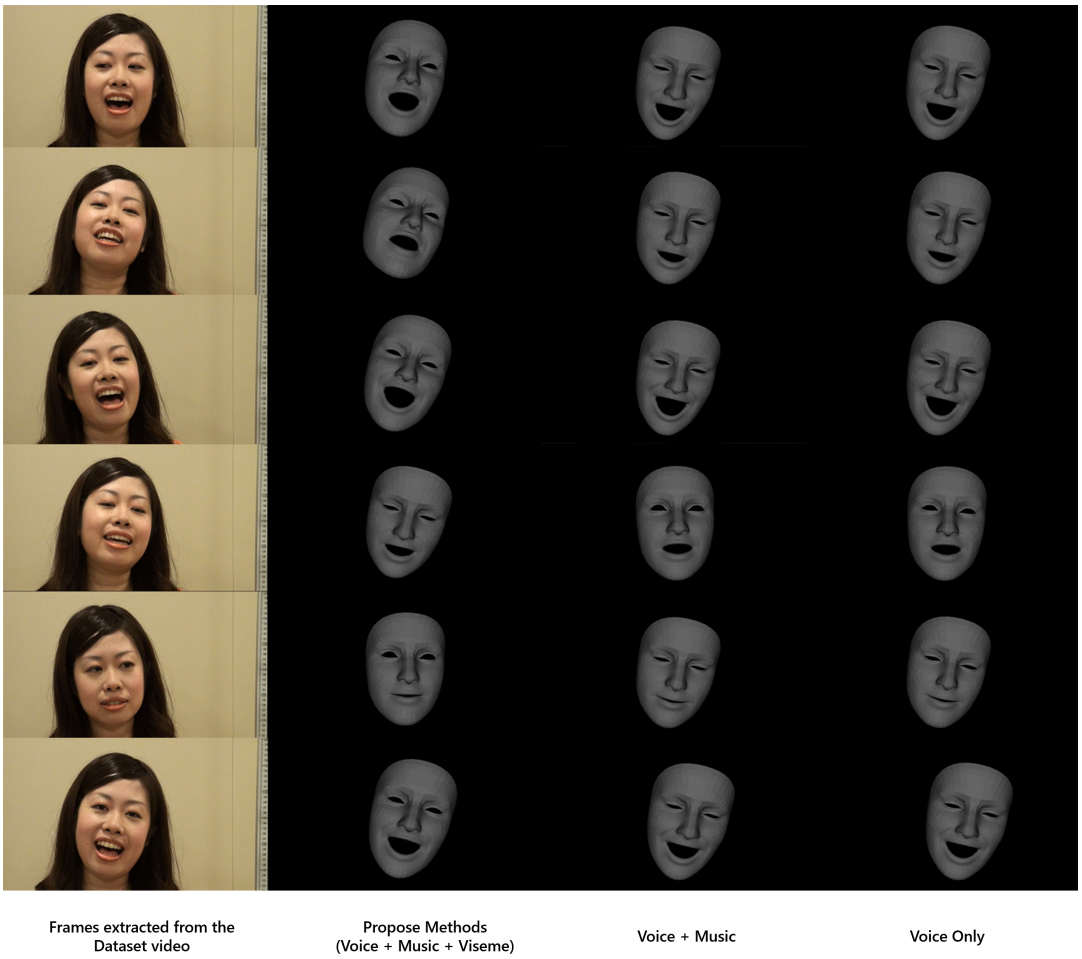


FIGURE 4.3: Comparison between the video from dataset, proposed method, proposed method without mouth shape feature and proposed method without mouth shape feature and music feature.

cameras. In addition, for each song, melody information and beat information manually created by the method of Goto et al., 2011 and others were created and used as song information. Results were generated with singing data of one song acquired from the same singer as training data and singing data of the other song as test data. From the singing information of another song, learn the behavioral characteristics at singing and model the behavioral features of singers singing as observed by certain songs. Using this learned model, an expression animation when the same singer sang another song is generated using singing information of the test data, music information, and mouth shape information. By comparing the measured test data in the data set with the weight coefficient of each facial expression outputted and the head rotation information, we verify how well the learned behavior characteristics

of the singer can be modeled.

The output data was converted to seven-dimensional output data using NMF, the batch size was 200, the number of LSTM lookback frames was 20 frames, the number of hidden layers of LSTM was 64, the number of learning epochs was 200. For calculation, NVIDIA GeForce 1080 Ti, GPU was used, and the learning time for one song was 13 minutes 21 seconds on average.

Fig. 4.3 shows the result of comparing the generated singing motion applied to the face model and the measured expression data. In addition, for the same LSTM model, we created results with only singing voice information as input, singing voice information and song information input as well, and show the expressions in the same frame side by side. Looking at the expressions, when comparing singing voice information, singing voice information and song information generated at the time of measurement, closing the eyes or putting power in the vicinity of the eyebrows, the mouth shape information, the proposed method achieved the most similar result. Focusing on the head rotation information, although the face was moved largely in the image at the time of measurement when the singing was performed, but the movement with the result generated by inputting only singing voice information, singing voice information and music information are relatively small. In the proposed method with mouth shape information added, it was possible to generate a state in which the face is rotating much like the image at the time of measurement.

Also, using the expression model created by Yamaguchi's method Yamaguchi et al., 2018, the result of applying the method to the high definition model is shown in Fig. 4.6. Approximately 51 blend shapes used in learning were prepared for facial expression models estimated from a single facial image by the method of Yamaguchi et al. And the facial expression parameters and head rotation information estimated by the proposed method were applied. Even for such a high-resolution face model, we can create expressive animation using the same method, and we confirmed the effectiveness of the animation generated by this method.

4.6 Evaluation

In this section, we will discuss about facial expression animation created by the proposed method. In order to show the effectiveness of the proposed method, numerical evaluation with measured expression data is performed as numerical precision evaluation. In addition, regarding the accuracy of modeling of naturalness and individual movement, subjective evaluation experiments were conducted on results generated using different input data, and the impression of the generation result was evaluated.

4.6.1 Numerical Evaluation

First, numerical evaluation is performed on facial expression animation generated by the proposed method. In this experiment, data of two songs measured from the data of three semi-professional singers mentioned in Section 5 are used. For each piece of singing data, learning was performed with data of one piece of music measured from the same singer, and a result was created based on the data of the other piece of music. For quantitative evaluation, animation is created with different input parameters for each. In the proposed method, singing voice information, song information, and mouth shape information are generated as input, but by comparing with the result generated by changing input information, validity of input information and learning method used in the proposed method . Specifically, we compare input data from the proposed method with no compression by NMF, one with singing voice information and song information as input data, and one with singing voice information as input. For the learning model, the learning parameters, and the calculator, all animation was learned and generated under the same conditions.

As a quantitative evaluation, the facial expression parameter and the head rotation information other than the output mouth are compared with the facial expression parameter and the head rotation information other than the measured mouth. For each generated animation, the expression parameter and the head rotation information at that time are compared by the mean square error with the expression parameter and the head rotation information acquired when actually measuring. We

TABLE 4.1: Quantitative evaluation using different input

Input Parameter	Full Data	A	B	C
Sing + Song + Mouth (NMF)	81.93	83.24	80.71	80.84
Sing + Song + Mouth(No Compress)	102.54	105.62	102.43	99.57
Sing + Song	88.88	90.42	89.11	87.11
Sing	90.77	94.15	88.23	89.93

calculated the squared error average for each singer and the squared error average of all the created data and compared and studied.

As written in the results in the table 4.1. A, B, and C represent different singers and indicate the average value of all the data. It can be seen that the proposed method in which the singing voice information, the song information, and the mouth shape information are dimensionally compressed to the output can reproduce the correct data with high precision as compared with the result by other inputs. In particular, since the accuracy is greatly improved even compared with the result when NMF dimension compression is not performed with the same input information, it can be said that dimensional compression by NMF contributes greatly to improvement of learning performance.

As a result of using only singing voice as an input, when compared with the result with singing voice information and song information as input, it can be seen that adding the mouth shape information greatly improves the result. Compared with only sound information such as singing voice information and music information, it is considered that better accuracy could be obtained by including facial expression information.

4.7 Subjective Evaluation

In order to evaluate the naturalness of the generation result and the behavior characteristic unique to the singer, three subjective evaluation experiments were conducted on the generated result. The screen used for the subjective evaluation experiment is shown in Fig. 4.4. In each question, subjects make a pair-wise comparison between the measured expression data and the results generated under various conditions. There were a total of 45 subjects, 34 men and 11 females, the age range from 20s to 40s, of which 6 are professional CG artists. In this subjective evaluation experiment,

a total of 6 animations were created for each of 2 types of patterns of three kinds of learning data. In each of the three patterns, the properties of input learning data are different, therefore expected results are different. First, the first pattern is to compare the results of learning with different singers and different songs and measurement data. In this pattern, test data with input data greatly differs in environments during learning and testing are obtained. We evaluate the stability of the learning model by evaluating whether natural motion can be learned when using such singing voice information as given input. Next, the second pattern is the result of learning and performing with different songs of the same singer. It is expected that similar results to the measured data will be obtained because it is a result learned from the same singer, although it is different songs. From this result, we verify how well the singer's singing behavior unique to the singer can be learned from the characteristics of the singing voice information of the singer. And the third pattern is the result of learning with the same song by different singers. Since it is a result of learning from the data of the same song but different singer, it is expected that results not similar to the measured data will be obtained. From this result, we verify whether we can learn the unique motion characteristics of different singers among the same songs.

4.7.1 Naturalness of the synthesized results

For the first pattern, we compared which of the measured data and the generated data seems to be natural as a movement of human beings by a 7-level Likert scale. Here, it is set that selecting 1 answers that measurement data is more natural and selecting 7 answers that generated result to be natural. The average value of the answer results is shown in the table 4.2. The average value of the answer result was 2.78, and it indicates that the result that the measured data seems more natural. On the other hand, looking at the histogram of the number of votes obtained in the answer result shown in Fig. 4.5 a., 22.2 % of subjects can not distinguish whether the result "cannot be chosen" or "the result is natural". Even in the generation results when the learning time and the testing time are greatly different, it can be stably generated the generation result that the number of subjects answering that the generation result is more natural than the measurement data has been indicated.

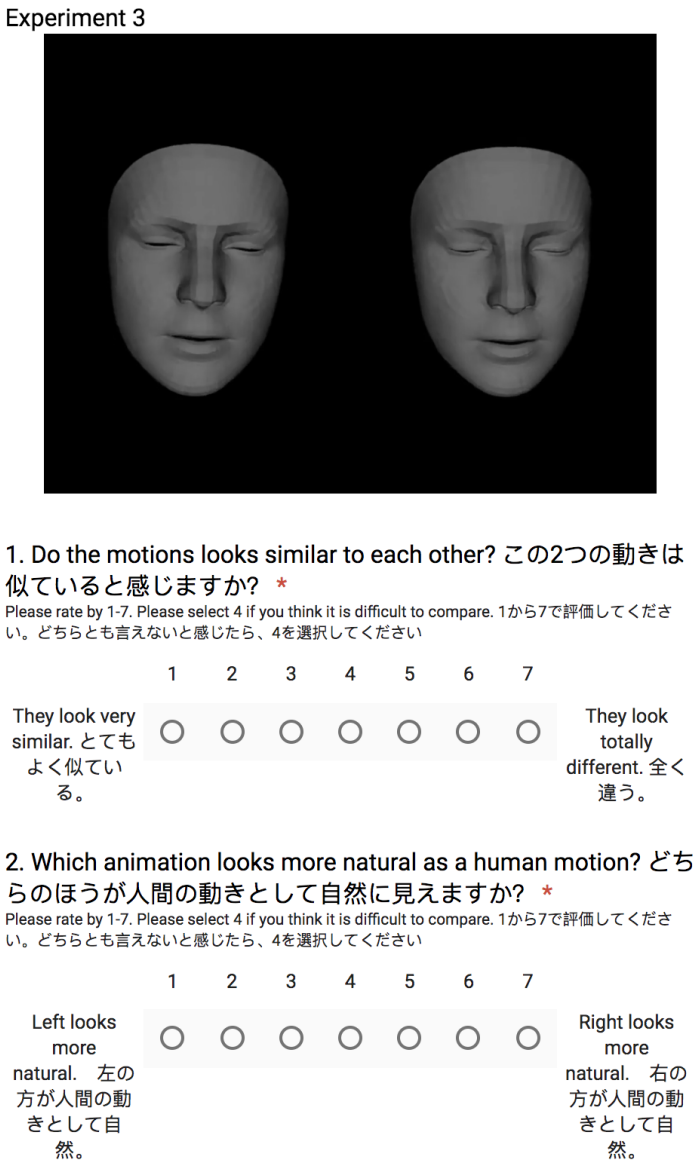
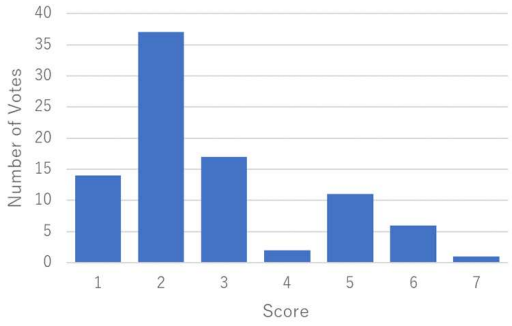
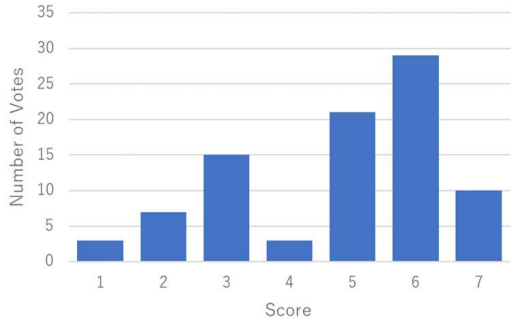


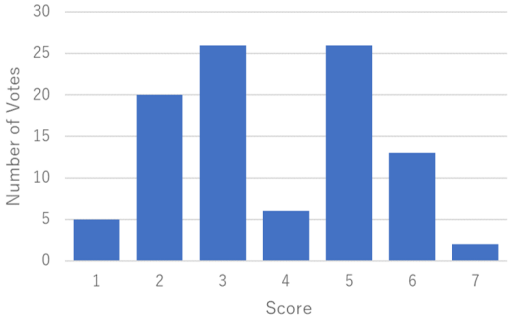
FIGURE 4.4: Subjective Experiment Interface



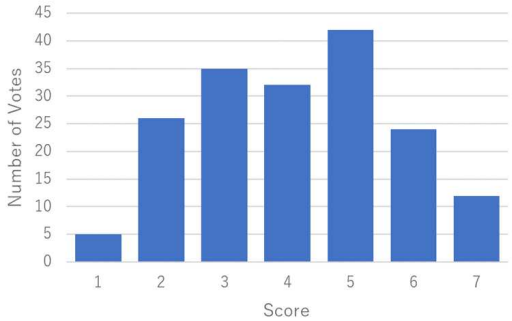
a. Naturalness evaluation compared with the grand truth when trained using different singer + different song



b. Naturalness evaluation compared with the grand truth when trained using different singer + same song and same singer + different song



c. Unsimilarity evaluation compared with the grand truth when trained using same singer + different song



d. Unsimilarity evaluation compared with the grand truth when trained using different singer + same song

FIGURE 4.5: Voting result of subjective Experiment

For the second pattern and the third pattern, we compared the measured data with the generated data, by selecting which one seems to be natural as a human motion using the 7-level Likert scale. The average value of the answer results is shown in the table 4.2. The average value of the answer result was 3.56. As you can see from the histogram on the number of votes obtained in the answer result shown in Fig. 4.5 b., which shows that the result is very close to the median value of 4, the ratio of selecting measurement data and generated result is almost identical. It indicates that natural results can be generated even by comparison with measured data.

In addition, average values are shown in Table 4.3 only by evaluation results of professional CG artists. The average value of the first pattern was 2.57, the second pattern and the average value of the third pattern was 3.21. The result here also evaluates the measurement data as being more natural if both the singer and the song are different, whereas if either one of the singer and the song is the same, it is close to the median value, and it was evaluated that it is natural as compared with measured data.

From the above results, it was found that the proposed method was able to generate natural results in pair-wise comparison with measured data when either singer or song was the same for learning and testing. Even under the harsh conditions that both singers and songs differ between learning and testing, since professional CG artists have found that they are able to produce natural results, It can be said that the proposed method is stable and it succeeded in generating a natural result to.

4.7.2 The individual expressivity of the result

For the second pattern, the dissimilarity between the measured data and the generated data was evaluated by the 7-level Likert scale. Table 4.2 shows the average value of the result. When both of the results seemed very similar to the measurement data the subjects are asked to select 1, and 7 is for not similar to the measurement data. In the second pattern, since it is the result of learning with different songs of the same singer, if it should be similar to the measurement data. If it succeeded, the proposed method models more accurately and distinctive behavior It can be said that it succeeded. The result of this experiment has an average value of 3.19, and it

was able to obtain the result that it is slightly similar. Looking at the histogram of the number of votes obtained in the figure shown in Fig. fig: hist c., We can see that there are many subjects evaluating "similar" or "very similar" .

For the third pattern, dissimilarity between the measured data and the generated data was evaluated by a 7-level Likert scale. Table 4.2 shows the average value of the result. When both of the results seemed very similar to the measurement data the subjects are asked to select 1, and 7 is for not similar to the measurement data. In the third pattern, the result of learning with the same song of different singers which should look different from the measured data, the proposed method models the behavior more precisely and distinctively. It can be said that the learning result have succeeded. The average value of the result was 4.32, and it was able to obtain the result that the generated results are not similar. Looking at the histogram of the number of votes obtained in the answer result shown in Fig. 4.5 d. The subjects evaluating "different" or "completely different" are significantly large. It is understood that the subject evaluated positively has exceeded the number of people who evaluated negatively. In order to judge the significance in the difference between these results, *Student's t-test* was conducted using the result of the second pattern and the result of the third pattern. The result of two-sided test was 0.000019, and it was confirmed that there was sufficient significant difference in this subjective evaluation experiment.

In addition, the evaluation results of only professional CG artists in the same experiment are shown in table 4.3. The results show that the average value of the second pattern was 2.79 and the average value of the third pattern was 4.93. From the perspective of a professional CG artist, the second pattern was evaluated as "similar", the third pattern evaluated as "not similar", and these results has a significantly large difference with the subject who was not a CG artist. We evaluated this that the subjects who are not CG artists have very few opportunities to compare and observe the unique expression of characters compared to CG artists. Therefore, it should have been difficult to make an absolute assessment as to whether or not they are similar or not, and the results were closer to the median value which indicates the uncertainty of the subjects decision. In Fig. 4.5 d., the subjects who evaluated to be 3 to 5 totaled 64.4 % of the subjects, and similar considerations can be made from this

TABLE 4.2: Subjective Evaluation on the naturalness and the similarity with the dataset

Learning Data	Comp w/ GT	Unsim w/ GT
Different Singer + Different Song	2.78	N/A
Same Singer + Different Song	3.52	3.19
Different Singer + Same Song	4.63	4.32

TABLE 4.3: Subjective Evaluation on the naturalness and the the similarity by professional CG Artists

Learning Data	Comp w/ GT	Unsim w/ GT
Different Singer + Different Song	2.57	N/A
Same Singer + Different Song	2.87	2.79
Different Singer + Same Song	3.57	4.93

that the many subjects had difficulty in making judgments.

From the results above, it can be said that we succeeded in generating an animation close to the measurement data by modeling the movement of the same singer even for different songs. When learning the behavior of different singers of the same song, we found that it is possible to obtain a result that is significantly different from the results learned by the same singer, though it does not result in much different generation results. On the other hand, professional CG artists' perspective that there was a significant difference in results, and professional CG artists found that the proposed method was effective both in creating natural motion and individual motions.

4.8 Conclusion

In this chapter, we proposed a facial expression animation generation method focused on singing structure and singing voice characteristics at singing. In order to estimate facial expression parameters other than the mouth shape and rotation component of the face, parameters related to the music were extracted from songs and singing voices, and mouth shape data peculiar to music was estimated on the mouth shape. By making learning of such data into a deep learning model taking time series into account, we succeeded in generating facial expression animation at high definition singing from elements that can be obtained from singing voices and songs.

As a future work, devising a learning model for songs with different tempos can be considered. At the moment, we are only experimenting when the tempo of

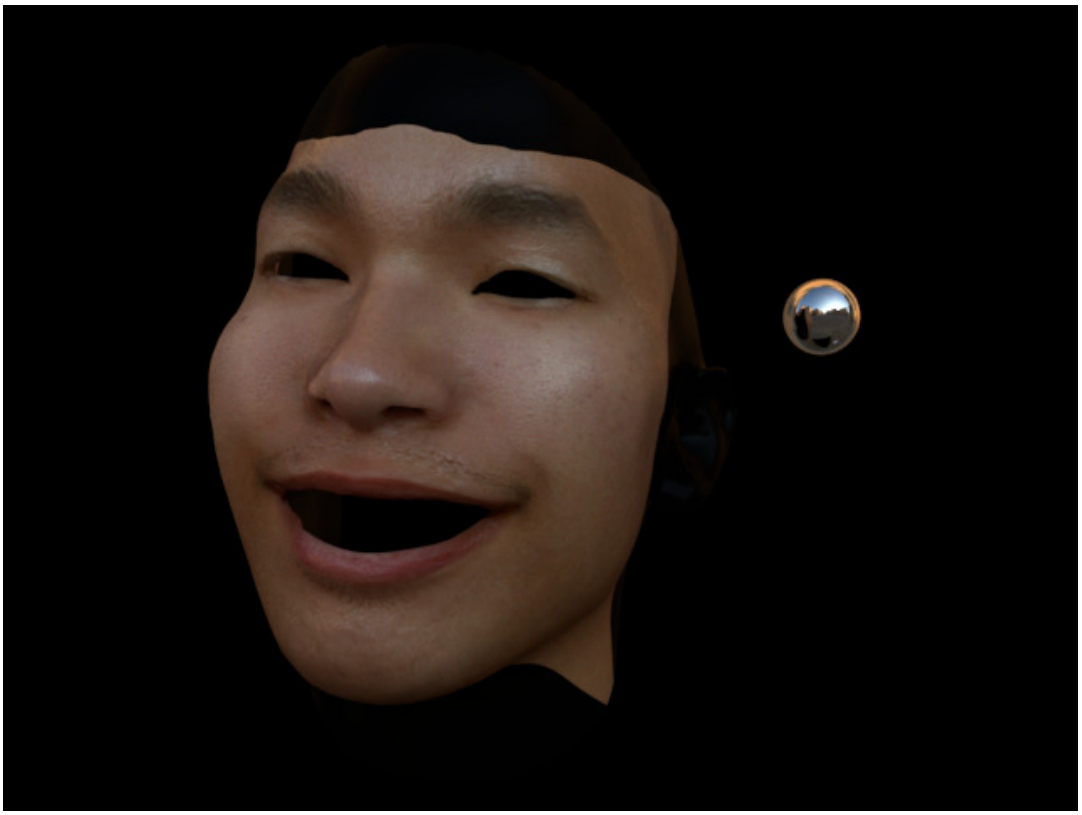


FIGURE 4.6: Singing Motion applied to photorealistic facial models.

the learning data and the test data are similar, and we have not done experiments with music with a large tempo differences. However, in the case of different tempos and songs, it is inferred that learning is difficult because the singing characteristics change significantly even for the same singer. In other researches using deep learning, there are examples that increase the generalization performance of learning models by increasing the number of learning data, and correspond to various kinds of inputs. Therefore, by constructing dataset at singing with various songs, we are able to devise a method that can deal with songs with wider variations of genre. Moreover, by gathering data of more singers, it is possible to estimate expressive features at singing from the facial expression features at the time of conversation and trying to generalize facial expressions in singing from the facial features of many singers Consider doing it.

Chapter 5

Real-time control interface for preserving the expressivity of the character

5.1 Introduction

In recent years, dance animation using digitally generated characters has been popular more than ever with it's growth of musical animation industry. As for the dance animation, one of the most popular way of creating such animation is to *pre-render* the dance animation. I can easily say that the pre-rendered animation can be useful for contents such as films or video where the viewers reaction does not play a large role, while real-time dance authoring has also been introduced due to its demands year by year. Novel sensors such as depth sensor cameras or the motion capture systems, the real-time control over the CG characters had become more accessible. These interfaces offered the user to create computer generated dance motions with intuitiveness. Though I are able to enjoy various live performances acted by CG characters, these live performances are created to be performed only to show the 2D video created in advance as an offline process. With the level of complexity in the backstage of live scene authoring, there lies the problem of purely in the skills of the performers but also the technical complexity to realize a high quality live stage performance. Accordingly, our goal in this chapter is to offers a character control system that requires a low level skills to not only control the character, but also seemlessly adapts into existing live stage performance frameworks.

The major challenge for the real-time dance authoring is that while some dance movements can be said that it is similar to normal motion, most of the dance moves are dissimilar in that its motion requires to follow the beat. In music and dance, the beat is defined as an audible or visual cue demarcating the separation of a certain

sequence of the music or dance. Dancers change their poses while continuously maintaining the beat of both the dance and music to fit to each other. The rhythm of musical content is normally defined by a repeating sequence of beats. Dance and music have evolved in tandem over centuries with the by having the beat in the middle to share and match them. Various existing motion control interfaces have not successfully been able to control high quality dance as these principles over the beat and rhythm are been ignored.

To address this problem, I considered a hypothesis that the interaction of real-time dance motion can be intuitive by allowing the users to understand the beat of both motion and music. The hypothesis has risen when observing the disk jockey in music. A disk jockey, known as a DJ, is a person who plays music by controlling the disk and sound interfaces to at a club or live performance. Their performance often involves intricate and seamless mixing to joint a piece of music with another, which is known as a way of performing music without playing musical instruments. Such interaction with the music normally considers the beat of the music to be the key to make the music continuously. Since both dance and music shares the similar principles of the beat, I expected that by providing semantics for the user to control the beat when controlling dance motion would allow the user a highly usability for system.

In this chapter, I present a real-time dance motion authoring interface based on such a hypothesis, called *DanceDJ*. By mimicking the interaction of the musical DJ in music, the system offers the intuitive control of the dance motion and create dance motion in real-time with high-quality. Implementation of the synchronize button in which I created to synchronizes the dance motion and the music played by the other electronic musical instruments, the user can easily match the beat of the dance motion to that of the music without losing the control over the character. In contrast to the DJ in music, the connectivity and beat of dance motion are much abstract than that of the music, which affects the intuitiveness during the time when user connects the dance motion to another. To support the users in such case, I have implemented a novel feature to compute the smoothness of the connection of the dance motions. The system computes the beat of the dance by using the motion intensity to estimate the probability of the frame wise connectivity of the dance motion, which represents

how well the beat of dance and music synchronize together. This dance specified feature offers the users a DJ-like experience when creating dance motion in real-time and provides better control to create the motion. Such interaction can be said that it realized our hypothesis of the beat and it's high correlation with music and dance motion.

We have built a fully usable real-time interface for dance authoring. I have conducted experiments in which the results indicated the capability of synthesizing high-quality dance motion in real-time using the proposed DJ interface. The experiments that I have conducted is the subjective evaluation for two perspectives: user and the audience. For the user, I have evaluated the level of usability for users to be able to create high-quality dance motion with intuitiveness. As for the audience, I have evaluated the quality of the created dance motions by watching the resulted dance motion in which our system has created in real-time at the stage performance.

There are two major contributions in this paper:

- Designing a novel DJ interface for real-time dance authoring based on dance and music correlations.
- Introducing a novel dance and music beat evaluation function in which they evaluate the quality of a pose in a dance synchronizes to that of another dance by the estimated beat of motion and music.

The rest of this chapter is organized as follow. Section **??** describes the related works of this project. Section **??** describes the DJ interface I have implemented in order to achieve real-time dance authoring. Section 5.4 describes the implementation of the transition function that estimate the quality of transitioning from a motion to another based on the pattern of the beat. Section 5.5 describes the way to visualize the results of the transition function in order to control the dance effectively. Section 5.6 explains the user study results to evaluate our system on how well it supports the usability of the system. Finally, Section 5.8 concludes and discusses the system.

5.2 Related Work

In this section, I introduce some of works that relates to our research. Numerous related works exists for creating character motion interactively both in HCI and CG

research field. Our related work can be clustered into two different categories; user interaction of character motion and DJ-like interactions.

5.2.1 Character Control User Interfaces

Various interfaces have been introduced to control character motion. One of the favorite ways is the performance-based approach. The interface, which uses motion capture and video of human motion has been widely used when creating 3DCG animation with dance motions. Ishigaki et al., 2009; Lee et al., 2002; Shiratori and Hodgins, 2008; Dontcheva, Yngve, and Popović, 2003; Fender, Müller, and Lindlbauer, 2015. Depth camera based motion sensing opens another pathway for real-time character controls Shum et al., 2013; Liu et al., 2017; Zhang et al., 2014; Liu et al., 2016. These interfaces fulfill the demands of creating accurate human motion, they relies fully upon the ability of the users to perform the motion the user want to create. Moreover, these interfaces are not capable of editing to create the final character motion. If the user wished to edit the data from the interface, the users are required to use other interface, such as sketch-based interaction Choi et al., 2016; Guay, Cani, and Ronfard, 2013; Jin et al., 2015; Hahn et al., 2015; Choi et al., 2012. While these interfaces are intuitive and easy to use, it is unrealistic to use such interfaces for creating real-time motion. Interfaces that control the bones of the characters Yoshizaki et al., 2011; Held et al., 2012; Zhai and Milgram, 1998; Jacobson et al., 2014, such as ones method proposed by Glauser et al., 2016, offers users a intuitive creation of the dance motion. These character-shaped interfaces that in which they deform the characters' bones allow users to make arbitrary poses with high interactiveness. Despite of being intuitive in creating body shapes for a specific frame, these interfaces are also not sufficient of controlling character motion in real-time, and it is unrealistic to use only two hands to create the animation with this interface. Similar to video games controls, the motion coordination keys in which the individual buttons are set with specific motion are widely used to create character motions Yazaki et al., 2015. In this way, the users are only required to press various buttons and achieve real-time control over the character movement. I can say that the usability is fine since with the observation that the multiple video games platform applies the same interface, though it is required to memorize the various motions allocated to each

particular button. This forces either the users to memorize a numerous patterns of button allocated motions or the motion creators to limit the variety of the motions allowed the character to be created.

5.2.2 DJ-like Interaction in Research

Due to its creative nature of mixing, various types of research have been focused upon the interaction of DJ's. The interaction of DJ's has been studied in many ways to create similar applications or improve the actions of them. While many of these applications introduces a novel interface alternative to the traditional DJ interface, the form of DJ interfaces has not changed significantly. Therefore, recent researchers have applied the DJ interaction to other applications in other domains to take an advantage of its high usability Ragnhild et al., 2003; Norman and Amatriain, 2007; Groth and Shamma, 2013. Target applications of DJ-like interaction varies from data visualization, 3DCG visualization to Robot motion control. In these researches, sliders and turntables have been installed in various real-time control applications. These research have proven that DJ-like interactions have capability of increasing the usability and provide the user with a novel experience of interaction with the target application in different domain, especially for the real-time controls.

Our user interface is inspired by the work done by Shirokura et al., 2010 which is called "Robot-Jockey". This interface introduced a novel interaction for controlling the dance motion of a robot using the DJ-like interaction of changing tempo and motion in real-time. Since the primary goal of this interface is to provide the means of creating the robot's motions in real-time with intuitive interaction, the interface only requires to select very limited motions options, for instance kicking or punching from the motion database. The limited selection of motions can be sufficient for a robot which only has a very few degree of freedom to control the body, while 3DCG characters have siginificantly larger number of the joints, bones and skin to take into account. Therefore, I follow the idea of controlling dance motions in real-time using a DJ-like interface while I improve the system to be sufficient to control 3DCG characters.

5.2.3 Implementation Requirements

Taking the shortcomings of related applications towards our potential users into account, I set our implementation requirements as follows:

Mixing interface for intuitive controls:

I select and control the parameters of the character motion that are necessary for the live-action remixing of dance motions.

Intuitive data transmission:

Our interface applied MIDI data transmission. The MIDI control system is a well-designed, very popular and intuitive interface in various fields that requires real-time interactions.

Real-time:

Our interface can create dance motions in real-time, which allows usage in live performances such as concerts.

In developing our interface, our goal was to implement all of the features into the interface which the works cited above have not yet completely satisfied. Since none of the related works and products have been able to fulfill the needs set out above, I believe that our interface that accomplished these goals is more advantageous for application in various situations.

5.3 DanceDJ Interface

In order to verify the hypothesis described in Section **??**, I developed the prototype of the interface to control the character's dance motion using a DJ interface. A live stage performance work flow is shown in Figure **??**. First, the DJ or the musician plays music along DanceDJ and shares the beat information with the DanceDJ system using a local network in real-time. While I have only evaluated our system playing along DJ, other electronic instruments and existing online beat estimation systems shares the similar data transmission system which make the proposed system to be applicable for wide variety of music genre. Secondly, the DanceDJ, as an operator not the system, controls an input dance motion to synchronize with the beat which is received from the musical instrument. At last, the created character

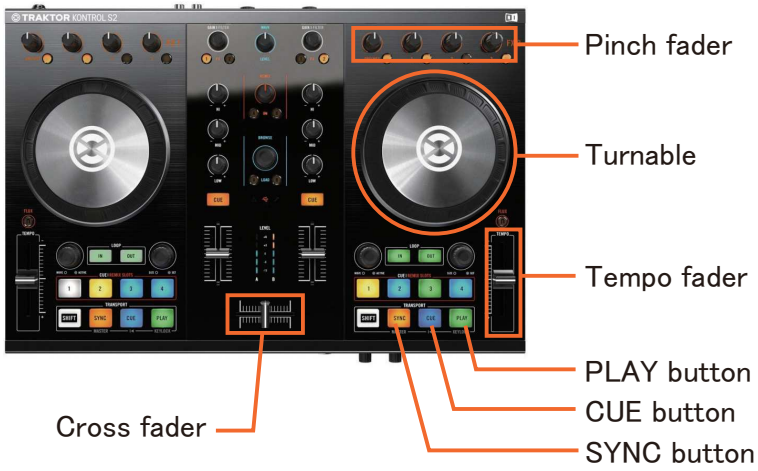


FIGURE 5.1: Common DJ Interface; buttons, fader and turntables are equipped for mixing two music.

with dance motion assigned by DanceDJ is projected on a screen or stage in front of audiences. From the perspective of the audiences, it appears to be like the character is dancing while maintaining the music played by the DJ. In our subjective evaluation experiment, the DJ used a DJ software package called *Traktor*, and sent music information to other device with a software called *Ableton Link*. The DJ interface and the DanceDJ interfaces shares a MIDI signal distributed at each button or slider to understand what the opposing player acting.

A typical DJ interface used in music controls is shown in Figure 5.1. This interface has various buttons and sliders to control music. The left and right part that have similar buttons and sliders are for controlling two different music respectively. The play button is for playing/pausing the music. The tempo faders are capable of adjusting the speed of the tempo on each music. The turntables interface in the middle can adjust playback speed to sync the tempos or beats. In the middle, the cross fader blends the two of music assigned to the left and right parts. The sync buttons are used to automatically sync the music tempo based on the other assigned music.

To control the character's dance motion, I map the different dance motion features controls functions onto the DJ controller as shown in Table 5.1. The idea behind such mapping design is to mimic the semantics of the buttons such that musical DJs can transfer their skills from music controls to dance controls. The tempo faders are allocated with a motion speed control function. The play buttons start and pause a dance motion. The turntables are used to playback sequences of dance motion

TABLE 5.1: Mapping Function between DJ and Dance Parameters

DJ-Interface	Dance control parameters
Tempo fader	Motion speed
Cross fader	Mixing dance motion
Pinch fader	Sound effect
Turntable	Moving in few sequence
PLAY	Start/Stop dance animation
SYNC	Synchronization between music and dance
CUE	Visualization of the transition point

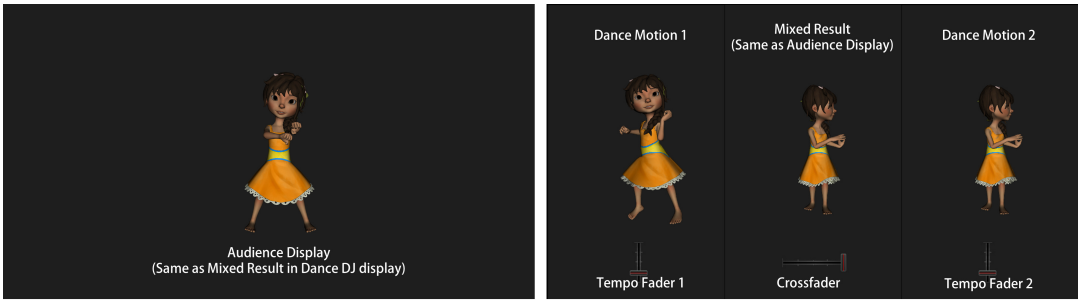


FIGURE 5.2: DanceDJ's system screen in a prototype stage. (a) For audience, the center's character displayed for a user is only projected to the projection screen. (b) For a user, the system displays three characters, which left and right are assigned different motions respectively, and center is a result interpolated dance motions between left and right by using cross fader.

frame by frame. The cross fader interpolates two different dance motions assigned in the either of the sides of the interface. The sync buttons are used for automatically tunes the tempo of the selected dance motion based on the music tempo received from the musical instruments. As for the cue button, I set alternative function from the musical DJ set up in which the button is used for the visualization function as described in Section 5.4 to find a smooth transition frame between each sides of the dance motions.

In Figure 5.2, I show a simple DanceDJ's system screen. From the user's perspective, the screen has three columns displaying the same character; the left and right columns display two different dance motions, and the center column displays the result that is interpolated from the left and right dance motions dependent on the cross fader's value. The audience can only see the resultant character in the center column.

Our system employs a data-driven approach to synthesize a new dance motion. I constructed a dance database that consisted of 16 dance motions from a Japanese video website *niconico*. In our study, this number of dance motions were sufficient for our experiments and a live performance for about half an hour. The dance motions were created by various amateur users, and the duration was about 3 minutes each. All dance motions are retargetted to the same structure with the same number of joints, which facilitates efficient motion interpolation. In our experiment, I used motion data with 70 joints included finger joints. Joints are represented using quaternions and I use *Spherical Linear Interpolation (SLERP)* for interpolating the joints between two dance motions.

It is a challenge to mix motions seamlessly and avoid sudden jumps of postures. While a DJ is able to mix different music by ad-lib as they have memorized the music, requiring the DJ to memorize all the dance choreography for dance mixing would be difficult. I address the problem by proposing a technical solution and by providing visual guidance. For the technical solution, I propose a novel transition function to evaluate a connectivity between two different dance motions based on the motion tempo, postures and the original music tempo assigned to each dance motion, which is detailed in Section 5.4. As the visual guidance, I visualize a result of the transition function to users on the system screen in real-time, which is described in Section 5.5.

5.4 A Transition Function for Dance Motions

In this section, I describe on a design for the transition function in order to evaluate how smoothly a frame in a dance can transit to a frame in another dance. This function supports a user to provide sufficient way to select the transition frame candidates of a selected dance motion. The function consists of two terms including (1) the beats of both music and motion $E_{beat}(i,j)$, and (2) the similarity of human poses $E_{pose}(i,j)$:

$$E_{trans}(i,j) = w_1 E_{beat}(i,j) + w_2 E_{pose}(i,j), \tag{5.1}$$

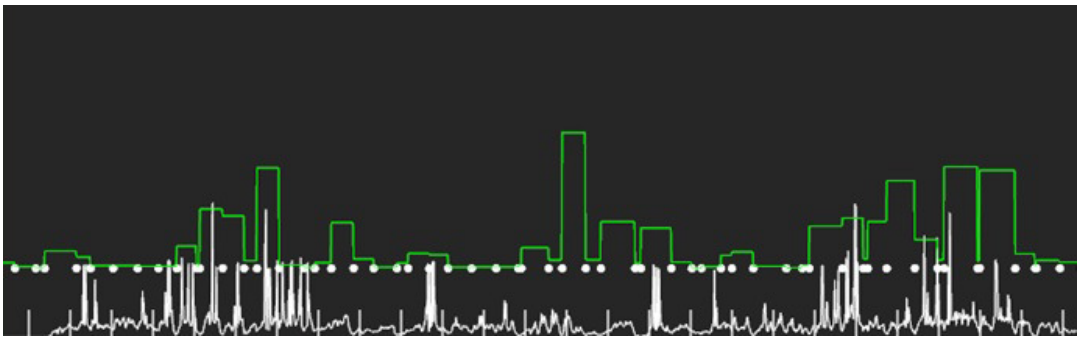


FIGURE 5.3: An analyzed result for arbitrary dance motion. The wave information is *Weight Effort (WE)* considered the sum of angular velocity for the all joints at each frame. Dot circles represents dance motion beats calculated from the *WE* value

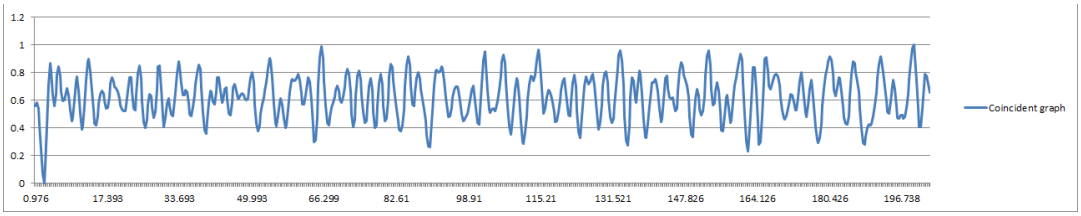


FIGURE 5.4: The beat evaluation result (E_{beat}) of transition function applied for arbitrary music and the dance motion created by artists. When the music beat and the motion beat are completely matching, the result becomes a constant sine curve.

where i and j are the beat indexes of two different dance motions, w_1 and w_2 are weights from 0.0 to 1.0 for the evaluation functions $E_{beat}(i,j)$ and $E_{pose}(i,j)$ respectively. These weight parameters are controlled by the user using the pinch fader of the DJ interface. $E_{beat}(i,j)$ and $E_{pose}(i,j)$ are described in the next two subsections.

5.4.1 Beat Matching between Music and Motion

Taking into account the fact that dance motion represents music beats as physical expression, the dance motion beat has correlation with joint angular velocity or angular moment. Accordingly, I first compute the *Weight Effort* using a sum of angular velocity for each dance motion in the database. I then define motion beat from the minimum value in each a certain window range Shiratori, Nakazawa, and Ikeuchi, 2006. Since the dance motion mostly comes with a corresponding music, I acquired the window range from the beat of the music.

As for the beat of the corresponding music, I applied the *Songle* API to analyze . The beat information in *Songle* API comes with a set of both time position and beat

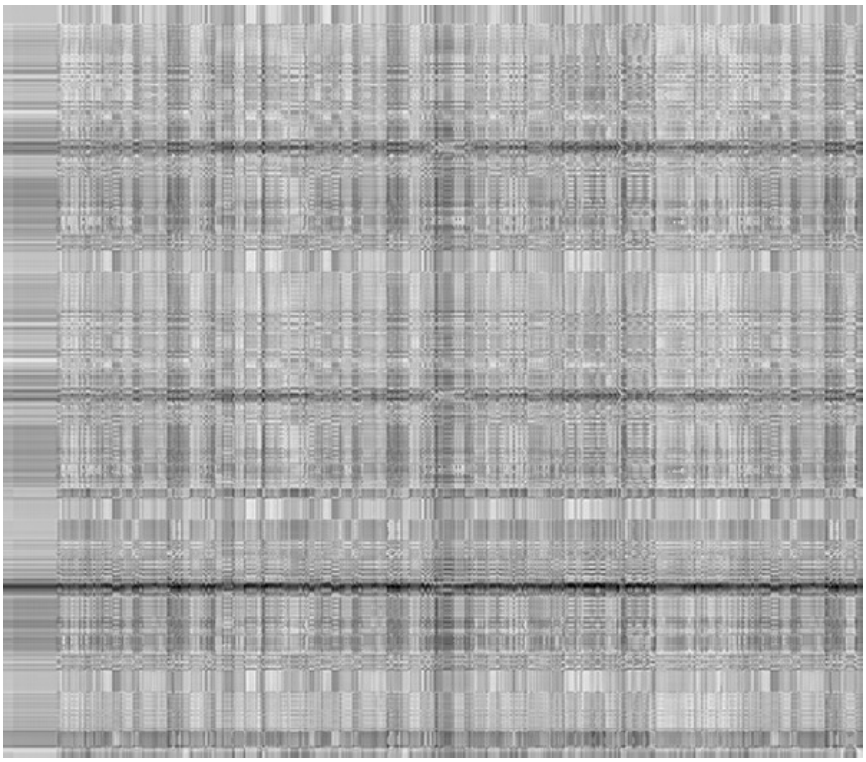


FIGURE 5.5: A result of *Motion Graph* applied for two different dance motions; The width and height of the image corresponds to the number of frames for each two dance motions. The corresponded frame colored with white represents high similarity of two postures.

count, and I calculate the beat per minutes (BPM) by calculating the average beat over time.

As shown in Figure 5.3, I observed that the dance motion beat does not always corresponds to the music beat because the dance motion beat tends not have constant interval as the musical beat. Our assumption was that a smooth transition of the dance should occur at the frame when the beat of both music and motion sync. Accordingly, I design the beat evaluation function $E_{beat}(i, j)$ to grade a beat corresponding rate between the motion and the music. To assess a beat's coincidence factor of motion tempo with music tempo, I estimate the discretized motion tempo into a continuous function using a Gaussian filter. The range of those weight values are from 0.0 to 1.0. I set 0.1 as the *sigma* value used in Gaussian distribution. The result is shown in Figure 5.4.

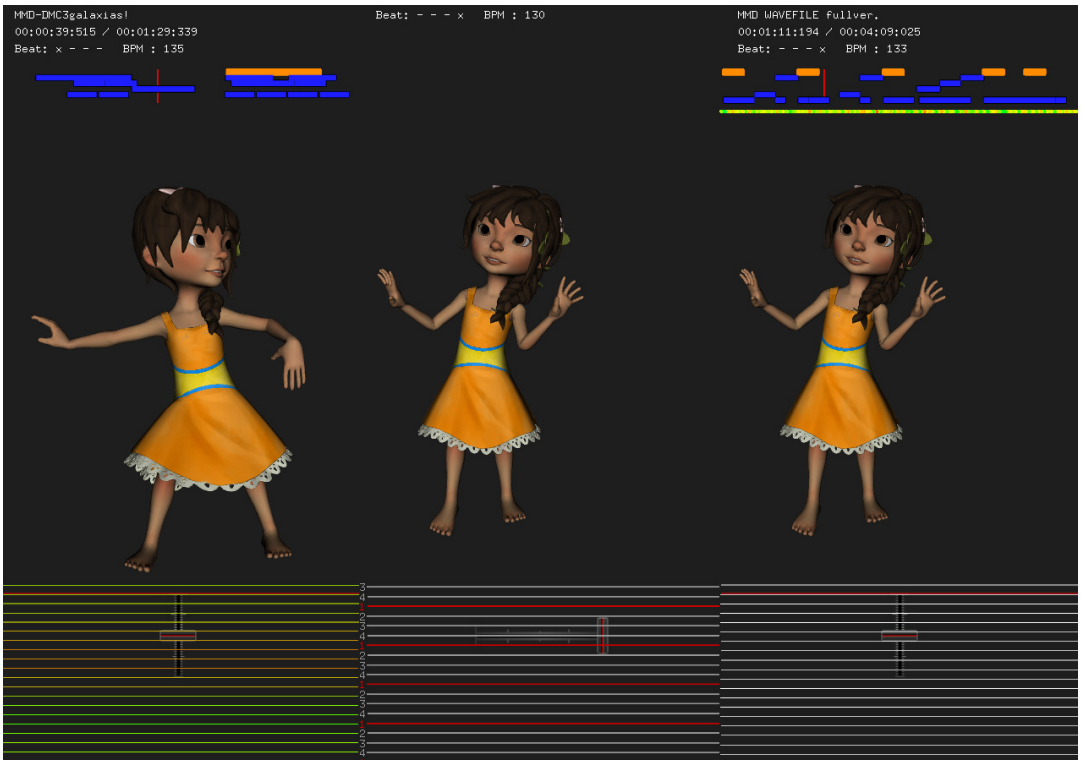


FIGURE 5.6: Final system screen; I show the beat information below. In the upper right hand corner visual guidance based on the transition function for seamless transition in shown.

5.4.2 Posture Similarity

To connect the different dance motions smoothly, I used the similarity of the pose as proposed in Kovar, Gleicher, and Pighin, 2002. In this method the similarity of the posture is define by the sum of root mean square distance in joint positions for all joints. Between two dance motions, the frame-wise pose similarity is calculated and represented using a similarity matrix as shown in Figure 5.5. The X-axis of the matrix represents the number of the frames in a dance motion, and the Y-axis of the matrix represents that of another dance. A darker the pixel is the more it indicates a high similarity at the corresponding frame pair.

We conduct this similarity computation for all dance motion in the dataset for every combinations as an pre-process. This has been done to reduce on-line computational time ruring the run-time. During the run-time, two poses from two dance motions are given and the calculated value is retrieved from the given similarity matrix to be used in the posture similarity function $E_{pose}(i, j)$.

5.5 Visual Guidance for Motion Transition

In this section, I explain on the visualization of the dance motion information on the screen UI to assist a user to achieve a smooth transition between two different dance motions.

An example of the graphical interface is shown in Figure 5.6. Our system has three regions on the system display; the left and right columns are for editing, and the center is for visualization for the mixing result. Once the user sets the cross fader to the right side, the user is able to select which dance motion the user would like to edit and start editing it on the left side.

Specifically, tn the top left and right hand corners of Figure 5.6, I allocated basic music and dance information which is the input dance motion, the music title, duration, beat, BPM and structure. The lines at the bottom left and right hand sides displays the music beat of the dance motion, which can be scrolled upwards while controlling the playback of a dance motion. The color on the line indicates the similarity level of the certain frames' transitioning calculated by solving Equation 5.1. At the bottom of the center column, the colored lines shows the music beat of four counts which the beat of the musical instrument sends through *Ableton Link*. When the user pushes the Sync button on the DanceDJ interface, the interval length of the music beat bars fits to the length of the beat received from the musical instrument.

5.5.1 Visualization of Transition Frames

Here,I describe a visualizing process for supporting the user during transitioning between different dance sequences.

When the user look for the next transition point by pushing the cue button, the system retrieves the smooth connection similarity data using Equation 5.1. A time-line in which it visualizes the results of the transition function can be seen below both the music structure. It shows that the time-line information which considers the global similarity (based on music and motion features such as tempo and pose) with the other candidate dance motion. Red has been used to represent large values and blue is to represent small values. Similarly, the color of the time-lines below the dancing character is keep on updated by the same color rule. It shows on the local

TABLE 5.2: Questions for audiences; (7-points Likert scale, 7: most agree; 1: least agree)

No.	Question
Q1.	Were you satisfied with the animation result?
Q2.	Is the connection between different dance sequences natural?
Q3.	Did you feel that the animation result matched both the dance and music?

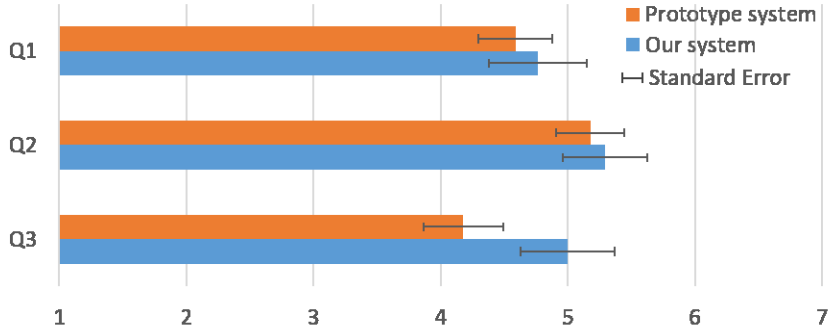


FIGURE 5.7: Result of the user evaluation from audience perspective.

similarity (based only on the pose similarity) between the currently selected dance frame and the other candidate's pose in the neighbor frame range.

5.6 User Study

In order to evaluate the effectiveness of our system, user studies have been conducted from the two perspective; audiences and users. 17 people (15 men and 2 women) and 12 people (11 men and 1 woman) has been involved in this experiment for the user and audience perspective studies respectively. Age range of the subjects are from 21 to 30 years old. None of the subjects had DJ experience, while 25 percent of the subjects had experience in dance or stage performance such as juggling.

5.6.1 Audience Perspective

In the study of audience perspective, two dance animations using our system synchronized with music played by a DJ was shown for a few minutes. The first animation was controlled by an experienced user without the transition function nor

automatic beat synchronization. From here I define this system as the prototype system. The second animation was controlled by the same user using the full range of functions that is proposed in this chapter. After showing both results, subjects were asked three questions for each synthesized animation shown in Table 5.2 to evaluate the visual effectiveness of the dance motion synthesized using the *transition function* in which we defined. For those three questions, the figure showed that our proposed method was better than the prototype stage which did not have the *transition function* and automatic beat synchronization. In other words, our method assisted the user in synchronizing the dance motion with the DJ's music and helped them to transition seamlessly between different dance motions. As for the dance quality from the audience's point of view, there were no significant differences between our proposed method and the prototype stage (p-values were more than 0.05 using a two-tailed Wilcoxon signed-rank test).

5.6.2 User Perspective

In the experiment of the user's perspective, I first instructed the basic usage of the prototype system to the subject for 5 minutes. The subject then used the system while listening to a DJ playing music for 10 minutes. Then, I explained our full system including the visualized transition function and automatic beat synchronized function for 5 minutes. The subject then used it for 10 minutes. In this way, the user can feel the difference of our proposed method compared with the prototype.

After the experiment, questions have been asked to each subject to evaluate the systems with five questions shown in Table 5.3. Figure 5.8 shows the averaged results of the user study. In Q1, Q2 and Q3, our system scores significantly overachieved the score of the prototype system. A two-tailed Wilcoxon signed-rank test was conducted to demonstrate that the difference between the average score was statistically significant. In Q4, majority of the subjects suggested that our system was more adaptable using the transition function between dance motion than the prototype system, as shown in Figure 5.9. In Q5, the transition function with the visual guidance offers subjects to intuitively control with small amount of time, as shown in Figure 5.10.

TABLE 5.3: Questions for users; Q1-Q4 (7-points Likert scale, 7: most agree; 1: least agree), Q5 (open ended)

No.	Question
Q1.	Could you naturally connect dance motion sequences?
Q2.	Could you match both dance and music?
Q3.	Were you satisfied with the dance animation you controlled.
Q4.	Did you feel that the mapping relationship between each button and motion function was adaptable?
Q5.	How long did it take you to control the dance motion satisfactorily?

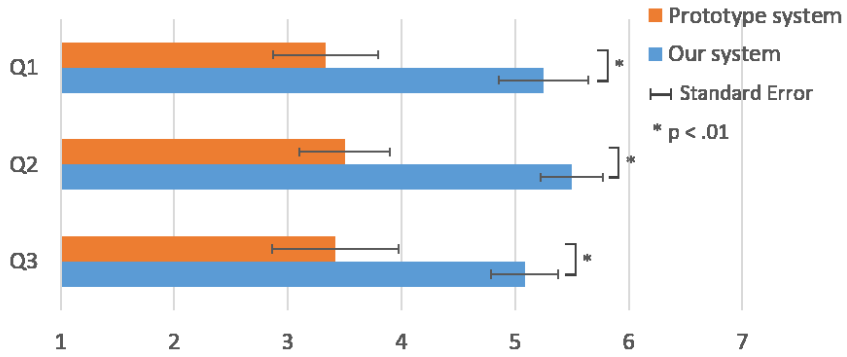


FIGURE 5.8: Result of the user evaluation from user perspective.

The results of user evaluation indicates that our method was more effective in transitioning seamlessly from both the audience and user's perspective.

5.6.3 Other Feedback

A part from the user studies above, a live stage performance with a DJ in a music club have been conducted. The synthesized character's dance motion was projected to a large 2D screen of the stage using a projector. 20-30 audience participated to our live performances. After the live stage performance, we conducted a simple survey to the audiences. The survey indicated that the positive reaction has been made from the audience when the character was synchronized with the DJ's music. During our live performance an opportunity to collaborate with VJ has been made who was getting ready switching the channel of the screen from ours to him. On the stage without being communication, the VJ offered us to mix two video channel half

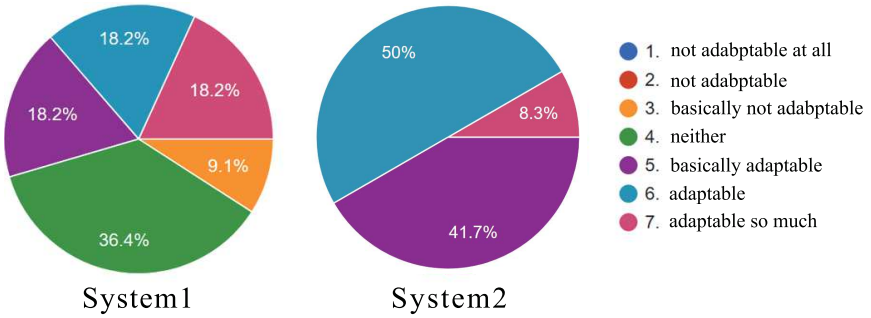


FIGURE 5.9: Question 4

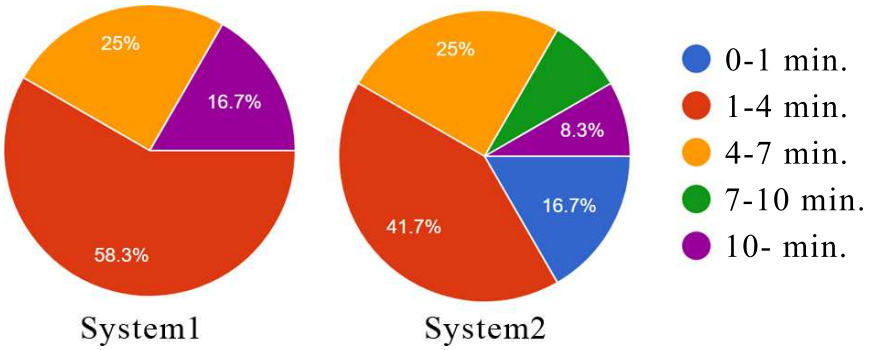


FIGURE 5.10: Question 5

and half, and the VJ made the visual effects such as the dance floor for our dancing character by ad-lib. This is shown in Figure 5.11. From such aexperience, we've felt an possibility of a collaboration with not only DJ but also VJ.

5.7 Discussions

Experimental feedback : As an evaluation, two experiments to evaluate both audience and user perspective. In order to evaluate the difference we set the experiments to set all the subjects to evaluate the prototype prior to evaluating our Dance DJ. Due to the fairness of our experiment, such comparison might not be sufficient. To counterbalance the evaluation more, an additional experiment in reverse order to evaluate detailed advantages of our system are required to be done. From our experiments, I found that DanceDJ system still has some rooms of improvement for both audiences and users. For instance, since our system can support user to seamlessly transit different dance motion naturally, sometimes many audiences do not even notice the timing of the transition. This raises the problem where the system may not

FIGURE 5.11: The scene of live performance. Left and right persons are DanceDJ player and DJ player. The middle person is next VJ player. Incidentally, DanceDJ player and a VJ player started a novel collaboration.

encourage drastically large motion effects at the transition. Such dramatic transition can be seen the stage for musical DJs when they transit between drastically different songs. As our future work, the improvemnt on the design in order to offer various connecting function, such as for a mash-up affection, which provide user intuitive control to create different varieties of dance motion with only simple change of the transition function. Some other future directions of the system are listed below:

Emotional Enrichment: Visualization of the emotional features of music in real-time. A character's facial and body motion are synchronized with music emotional features perceived by the audiences. Integrating existing techniques such as automatic facial motion synthesis Asahina et al., 2015 and motion filter Wang et al., 2006 using the pinch fader to enrich quality of the animation might add more expressiveness to the system.

Database Limitation: For all the experiments, a motion database created by armature artists has been made. Hence, the quality and the quantity of the motion database are very limited at this point. In the future, collecting more dance motions and corresponded music data-set from the video shared in different video hosting services is required using Computer Vision technique.

Helping Users with Disabilities: Since the system only requires the user to control a few function to create the dance motion, I have the expectation on the system to be able to help those who cannot dance due to physical disabilities. I am currently planning to evaluate how to better serve such a user group by providing this system to generate virtual dance characters for their enjoyment. We plan to integrate more functions for wider range of users with physical disability to be able to enjoy dance through our system. **Artists Collaboration**: I believe that the users of our system,

DanceDJs, have a high possibility to be a new type of major artists in the field, such as DJs or VJs. From our experiences in the collaboration with DJs as well as VJs has been successful, the collaborations with other kinds of visual artists will be a very interesting direction to move forward for exploring future possibilities.

5.8 Conclusion

In this chapter I have presented DanceDJ, a novel DJ-like interface to create dance animation in real-time. I have presented the transition function and visual guidance to support the user control as our main contribution. The transition function considered both synchronization between the music and human motion beats, and pose similarity with next dance's pose. By visualizing the transition function result, the system offers the user to intuitively and effectively control the transition to the next dance motion smoothly. Evaluation for the effectiveness from audience and user perspectives has been made and had positive reactions from the both sides.

As for our future work, the other features that might help the user experience will be introduce. A motion effect function for changing the dynamics of dance motions with using the pinch fader is one of the options. As for collaboration with Video Jockeys (VJs), I will build a system the dancing character can play with VJs and theyu control the background environment to be more attractively. Furthermore, our system will need to be able to find arbitrary dance motions from dance motion database as soon as possible as an on-line process by using a motion exploring function. I believe that DanceDJ has a high potential to become a new style of live performance in collaboration with DJ, VJ and real dancers on a stage.

Chapter 6

Conclusion

In this chapter we conclude this dissertation by clarifying the orientation of our research on musical animation synthesis. Furthermore, we discuss future work of the related fields.

6.1 Summary

We have described our work on musical animation synthesis specifically on facial animation during singing and the interface to control dance motion, in order to support artists to to create live stage performances. Through 3 chapters of our work(Chapter 2 to Chapter 4), we have demonstrated the glimpse of the musical animation synthesis field. Facial animation and the interaction of the musical animation field still requires numerous deep dives into the details, while this dissertation provided the beginning of the musical animation synthesis. These two fields and the the dance motion processing will eventually play huge roles not only for the character animation industry but also the music information synthesis to be the new corner stone of the fields.

6.2 The future of the fields

Ever since first live stage performance of the 3DCG avatar, similar performances have been conducted. Some of the performances have been extremely successful, while some others are not. The details of the difference in such similar stages is yet unknown, the fact tells us that the viewers are more and more capable of recognizing the high quality animation and low quality animations with the development of the industry. Additionally, as the animation quality develops, the uncanny valley of 3DCG animation will play significant roles on the viewers. As the current demands sustains, our research will be revisited in near future and wish to play bigger roles.

We are working on several live stage performance of the singer whom she is already passed away. During the creation some of the factors have been mentioned for the future of this field of study. In this section we will discuss some of the fields that this area of study lacks and the hints for the future.

6.2.1 The dataset creation for the musical animations

The introduction of machine learning revisited the importance in the high=quality data, and this field is no exception. While animation dataset is very small, the size of the musical animation is significantly low. The hopes of the fields is the video datasets of people singing on the streaming services. If many people have posted the data more often than not, we are able to require a large dataset of people singing or dancing. The large dataset of musical animation is not only important

6.2.2 The singing specific facial rigs

Our singing facial animation synthesis explained in Chapter3 has been implemented to the live stage performance to be held. During the time, the discussion had been made that the way of rigging the character when people sings seems to behave far apart from the daily facial animation. In speech animation the discussions have been made that the spcial rig specified for the mouth motion. Singing requires movements that is specified for singing such as the breathing, blinking, the feeling or the difference in genre of the song. To well represent such behavior it might be necessary to model the singing specific facial rigs. In order to find the best representation of the facial animation control rigs, the number of the dataset is significantly important. As being addressed in the previous section, creation of the dataset will also have high hopes for supporting this.

6.2.3 Realtime control of singing animation

In this dissertation, we have only provided a solution for realtime control on dancing. While singing facial animation does not require the large transition in motion compared with dancing, there are demands in controlling it in realtime. Facial animation control such as turning to the specific directions or change in feelings during

the singing would add more expression in the final animation. To add these expression naturally, not only the interface to control the facial expression but also the methods to extract the details of the expression and naturally insert them is required.

6.3 Outro

As being addressed in the Chapter 1. the live stage performances of the avatar already exists by the hard works of the artists. Although, the number of such stage is still low that it is still unknown what kind of techniques will help increase the reality and the attractiveness of the avatar. Finding aesthetics of the character will only be done after numbers of the stages to be conducted. The automation of the creating process will increase the number of such contents and more viewers get a chance to see the stages. The knowledge we will find in these stages will contribute to various similar fields.

We hope that our works will be foundation of the future development not only in the musical animation field but also other fields of the computer graphics and music information processing.

Bibliography

Alexander, Oleg et al. (2009). "The Digital Emily project: photoreal facial modeling and animation". In: *Acm siggraph 2009 courses*. ACM, p. 12.

Asahina, Wakana et al. (2015). "Automatic Facial Animation Generation System of Dancing Characters Considering Emotion in Dance and Music". In: *SIGGRAPH Asia 2015 Posters*. SA '15. Kobe, Japan: ACM, 11:1–11:1. ISBN: 978-1-4503-3926-1. DOI: 10.1145/2820926.2820935. URL: http://doi.acm.org/10.1145/2820926.2820935.

Bergeron, Philippe and Pierre Lachapelle (1985). "Controlling facial expressions and body movements in the computer generated animated short\'Tony de Peltrie\'". In:

Blanz, Volker and Thomas Vetter (1999). "A morphable model for the synthesis of 3D faces". In: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., pp. 187–194.

Bouaziz, Sofien, Yangang Wang, and Mark Pauly (2013). "Online modeling for real-time facial animation". In: *ACM Transactions on Graphics (TOG)* 32.4, p. 40.

Cao, Chen et al. (2013). "3D shape regression for real-time facial animation". In: *ACM Transactions on Graphics (TOG)* 32.4, p. 41.

Cao, Xuan et al. (2017). "Sparse Photometric 3D Face Reconstruction Guided by Morphable Models". In: *arXiv preprint arXiv:1711.10870*.

Choe, Byoungwon and Hyeong-Seok Ko (2005). "Analysis and synthesis of facial expressions with hand-generated muscle actuation basis". In: *ACM SIGGRAPH 2005 Courses*. ACM, p. 15.

Choi, Byungkuk et al. (2016). "SketchiMo: Sketch-based Motion Editing for Articulated Characters". In: *ACM Trans. Graph.* 35.4, 146:1–146:12. ISSN: 0730-0301. DOI: 10.1145/2897824.2925970. URL: http://doi.acm.org/10.1145/2897824.2925970.

Choi, M. G. et al. (2012). "Retrieval and Visualization of Human Motion Data via Stick Figures". In: *Comput. Graph. Forum* 31.7pt1, pp. 2057–2065. ISSN: 0167-7055. DOI: 10.1111/j.1467-8659.2012.03198.x. URL: http://dx.doi.org/10.1111/j.1467-8659.2012.03198.x.

Chuang, Erika S and Christoph Bregler (2004). "Analysis, synthesis, and retargeting of facial expressions". PhD thesis. Stanford University.

Dontcheva, Mira, Gary Yngve, and Zoran Popović (2003). "Layered Acting for Character Animation". In: *ACM SIGGRAPH 2003 Papers*. SIGGRAPH '03. San Diego, California: ACM, pp. 409–416. ISBN: 1-58113-709-5. DOI: 10.1145/1201775.882285. URL: http://doi.acm.org/10.1145/1201775.882285.

Ekman, Paul and Erika L Rosenberg (1997). *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA.

Fan, Rukun, Songhua Xu, and Weidong Geng (2012). "Example-based automatic music-driven conventional dance motion synthesis". In: *IEEE transactions on visualization and computer graphics* 18.3, pp. 501–515.

Fender, Andreas, Jörg Müller, and David Lindlbauer (2015). "Creature Teacher: A Performance-Based Animation System for Creating Cyclic Movements". In: *Proceedings of the 3rd ACM Symposium on Spatial User Interaction*. SUI '15. Los Angeles, California, USA: ACM, pp. 113–122. ISBN: 978-1-4503-3703-8. DOI: 10.1145/2788940.2788944. URL: http://doi.acm.org/10.1145/2788940.2788944.

Fukayama, Satoru and Masataka Goto (2014). "Automated choreography synthesis using a gaussian process leveraging consumer-generated dance motions". In: *Proceedings of the 11th Conference on Advances in Computer Entertainment Technology*. ACM, p. 23.

Glauser, Oliver et al. (2016). "Rig Animation with a Tangible and Modular Input Device". In: *ACM Trans. Graph.* 35.4, 144:1–144:11. ISSN: 0730-0301. DOI: 10.1145/2897824.2925909. URL: http://doi.acm.org/10.1145/2897824.2925909.

Goto, Masataka et al. (2011). "Songle: A Web Service for Active Music Listening Improved by User Contributions". In: *ISMIR*.

Groth, Paul and David A. Shamma (2013). "Spinning Data: Remixing Live Data Like a Music Dj". In: *CHI '13 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '13. Paris, France: ACM, pp. 3063–3066. ISBN: 978-1-4503-1952-2. DOI: 10.1145/2468356.2479611. URL: http://doi.acm.org/10.1145/2468356.2479611.

Guay, Martin, Marie-Paule Cani, and Rémi Ronfard (2013). "The Line of Action: An Intuitive Interface for Expressive Character Posing". In: *ACM Trans. Graph.* 32.6, 205:1–205:8. ISSN: 0730-0301. DOI: 10.1145/2508363.2508397. URL: http://doi.acm.org/10.1145/2508363.2508397.

Hahn, Fabian et al. (2015). "Sketch Abstractions for Character Posing". In: *Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation*. SCA '15. Los Angeles, California: ACM, pp. 185–191. ISBN: 978-1-4503-3496-9. DOI: 10.1145/2786784.2786785. URL: http://doi.acm.org/10.1145/2786784.2786785.

Held, Robert et al. (2012). "3D Puppetry: A Kinect-based Interface for 3D Animation". In: *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*. UIST '12. Cambridge, Massachusetts, USA: ACM, pp. 423–434. ISBN: 978-1-4503-1580-7. DOI: 10.1145/2380116.2380170. URL: http://doi.acm.org/10.1145/2380116.2380170.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). "Long short-term memory". In: *Neural computation* 9.8, pp. 1735–1780.

Huynh, Loc et al. (2018). "Mesoscopic Facial Geometry Inference Using Deep Neural Networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8407–8416.

Ishigaki, Satoru et al. (2009). "Performance-based Control Interface for Character Animation". In: *ACM Transactions on Graphics (SIGGRAPH)* 28.3.

Jacobson, Alec et al. (2014). "Tangible and Modular Input Device for Character Articulation". In: *ACM Trans. Graph.* 33.4, 82:1–82:12. ISSN: 0730-0301. DOI: 10.1145/2601097.2601112. URL: http://doi.acm.org/10.1145/2601097.2601112.

Jin, Ming et al. (2015). "AniMesh: Interleaved Animation, Modeling, and Editing". In: *ACM Trans. Graph.* 34.6, 207:1–207:8. ISSN: 0730-0301. DOI: 10.1145/2816795.2818114. URL: http://doi.acm.org/10.1145/2816795.2818114.

Joshi, Pushkar et al. (2006). "Learning controls for blend shape based realistic facial animation". In: *ACM Siggraph 2006 Courses*. ACM, p. 17.

Kaji, Shizuo et al. (2012). "Mathematical analysis on affine maps for 2D shape interpolation". In: *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. Eurographics Association, pp. 71–76.

Kakitsuka, Ryo et al. (2017). "Authoring System for Choreography Using Dance Motion Retrieval and Synthesis". In: *The 30th International Conference on Computer Animation and Social Agents (CASA 2017)*, pp. 122–131.

Kanade, Takeo, Ying li Tian, and Jeffrey F. Cohn (2000). "Comprehensive Database for Facial Expression Analysis". In: *FG*.

Kim, Hyeongwoo et al. (2018). "Deep video portraits". In: *ACM Transactions on Graphics* 37.4, 1–14. ISSN: 0730-0301. DOI: 10.1145/3197517.3201283. URL: http://dx.doi.org/10.1145/3197517.3201283.

Kovar, Lucas, Michael Gleicher, and Frédéric Pighin (2002). "Motion Graphs". In: *ACM Trans. Graph.* 21.3, pp. 473–482. ISSN: 0730-0301. DOI: 10.1145/566654.566605. URL: http://doi.acm.org/10.1145/566654.566605.

Lee, Jehee et al. (2002). "Interactive Control of Avatars Animated with Human Motion Data". In: *ACM Trans. Graph.* 21.3, pp. 491–500. ISSN: 0730-0301. DOI: 10.1145/566654.566607. URL: http://doi.acm.org/10.1145/566654.566607.

Li, Hao, Thibaut Weise, and Mark Pauly (2010). "Example-based facial rigging". In: *Acm transactions on graphics (tog)* 29.4, p. 32.

Li, Hao et al. (2013). "Realtime facial animation with on-the-fly correctives." In: *ACM Trans. Graph.* 32.4, pp. 42–1.

Liu, Z. et al. (2017). "Template Deformation-Based 3-D Reconstruction of Full Human Body Scans From Low-Cost Depth Cameras". In: *IEEE Transactions on Cybernetics* 47.3, pp. 695–708. ISSN: 2168-2267. DOI: 10.1109/TCYB.2016.2524406.

Liu, Zhiguang et al. (2016). "Kinect Posture Reconstruction based on a Local Mixture of Gaussian Process Models". In: *IEEE Transactions on Visualization and Computer Graphics* 22.11, pp. 2437–2450. ISSN: 1077-2626. DOI: 10.1109/TVCG.2015.2510000.

Livingstone, Steven R, William Forde Thompson, and Frank A Russo (2009). "Facial expressions and emotional singing: A study of perception and production with

motion capture and electromyography". In: *Music Perception: An Interdisciplinary Journal* 26.5, pp. 475–488.

Logan, Beth et al. (2000). "Mel Frequency Cepstral Coefficients for Music Modeling." In: *ISMIR*. Vol. 270, pp. 1–11.

Lu, Zhihe et al. (2017). *Recent Progress of Face Image Synthesis*. arXiv: 1706.04717 [cs.CV].

Norman, Alex and Xavier Amatriain (2007). "Data jockey, a Tool for Meta-Data Enhanced Digital DJing and Active listening." In: *ICMC*. Michigan Publishing.

Pighin, Frédéric et al. (2006). "Synthesizing realistic facial expressions from photographs". In: *ACM SIGGRAPH 2006 Courses*. ACM, p. 19.

Quinto, Lena R et al. (2014). "Singing emotionally: a study of pre-production, production, and post-production facial expressions". In: *Frontiers in psychology* 5, p. 262.

Ragnhild, Mark Mckelvin et al. (2003). "SeismoSpin: A Physical Instrument for Digital Data". In: *In CHI f03: CHI f03 extended abstracts on Human factors in computing systems*. ACM Press, pp. 832–833.

Saito, Jun (2013). "Smooth contact-aware facial blendshapes transfer". In: *Proceedings of the Symposium on Digital Production*. ACM, pp. 7–12.

Saito, Shunsuke et al. (2017). "Photorealistic facial texture inference using deep neural networks". In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*. Vol. 3.

Selim, Ahmed, Mohamed Elgharib, and Linda Doyle (2016). "Painting style transfer for head portraits using convolutional neural networks". In: *ACM Transactions on Graphics (ToG)* 35.4, p. 129.

Shiratori, Takaaki and Jessica K. Hodgins (2008). "Accelerometer-based User Interfaces for the Control of a Physically Simulated Character". In: *ACM Trans. Graph.* 27.5, 123:1–123:9. ISSN: 0730-0301. DOI: 10.1145/1409060.1409076. URL: http://doi.acm.org/10.1145/1409060.1409076.

Shiratori, Takaaki, Atsushi Nakazawa, and Katsushi Ikeuchi (2006). "Dancing-to-Music Character Animation". In: *Comput. Graph. Forum* 25.3, pp. 449–458. DOI: 10.1111/j.1467-8659.2006.00964.x. URL: http://dx.doi.org/10.1111/j.1467-8659.2006.00964.x.

Shirokura, Takumi et al. (2010). "RoboJockey: Real-time, Simultaneous, and Continuous Creation of Robot Actions for Everyone". In: *Proceedings of the 7th International Conference on Advances in Computer Entertainment Technology*. ACE '10. Taipei, Taiwan: ACM, pp. 53–56. ISBN: 978-1-60558-863-6. DOI: 10.1145/1971630.1971646. URL: http://doi.acm.org/10.1145/1971630.1971646.

Shoemake, Ken and Tom Duff (1992). "Matrix animation and polar decomposition". In: *Proceedings of the conference on Graphics interface*. Vol. 92. Citeseer, pp. 258–264.

Shum, Hubert P. H. et al. (2013). "Real-Time Posture Reconstruction for Microsoft Kinect". In: *IEEE Transactions on Cybernetics* 43.5, pp. 1357–1369. ISSN: 2168-2267. DOI: 10.1109/TCYB.2013.2275945.

Sumner, Robert W and Jovan Popović (2004). "Deformation transfer for triangle meshes". In: *ACM Transactions on Graphics (TOG)*. Vol. 23. 3. ACM, pp. 399–405.

Suwajanakorn, Supasorn, Steven M Seitz, and Ira Kemelmacher-Shlizerman (2017). "Synthesizing obama: learning lip sync from audio". In: *ACM Transactions on Graphics (TOG)* 36.4, p. 95.

Tena, J Rafael, Fernando De la Torre, and Iain Matthews (2011). "Interactive region-based linear 3d face models". In: *ACM Transactions on Graphics (TOG)*. Vol. 30. 4. ACM, p. 76.

Thompson, William Forde, Phil Graham, and Frank A Russo (2005). "Seeing music performance: Visual influences on perception and experience". In: *Semiotica* 2005.156, pp. 203–227.

Thompson, William Forde, Frank A Russo, and Steven R Livingstone (2010). "Facial expressions of singers influence perceived pitch relations". In: *Psychonomic bulletin & review* 17.3, pp. 317–322.

Vlasic, Daniel et al. (2005). "Face transfer with multilinear models". In: *ACM transactions on graphics (TOG)* 24.3, pp. 426–433.

Wang, Jue et al. (2006). "The Cartoon Animation Filter". In: *ACM Trans. Graph.* 25.3, pp. 1169–1173. ISSN: 0730-0301. DOI: 10.1145/1141911.1142010. URL: http://doi.acm.org/10.1145/1141911.1142010.

Weise, Thibaut, Bastian Leibe, and Luc Van Gool (2007). "Fast 3d scanning with automatic motion compensation". In: *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, pp. 1–8.

Yamaguchi, Shuco et al. (2018). "High-fidelity facial reflectance and geometry inference from an unconstrained image". In: *ACM Transactions on Graphics (TOG)* 37.4, p. 162.

Yazaki, Y. et al. (2015). "Automatic Composition by Body-Part Motion Synthesis for Supporting Dance Creation". In: *2015 International Conference on Cyberworlds (CW)*, pp. 200–203. DOI: 10.1109/CW.2015.26.

Yoshizaki, Wataru et al. (2011). "An Actuated Physical Puppet As an Input Device for Controlling a Digital Manikin". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '11. Vancouver, BC, Canada: ACM, pp. 637–646. ISBN: 978-1-4503-0228-9. DOI: 10.1145/1978942.1979034. URL: http://doi.acm.org/10.1145/1978942.1979034.

Zhai, Shumin and Paul Milgram (1998). "Quantifying Coordination in Multiple DOF Movement and Its Application to Evaluating 6 DOF Input Devices". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '98. Los Angeles, California, USA: ACM Press/Addison-Wesley Publishing Co., pp. 320–327. ISBN: 0-201-30987-4. DOI: 10.1145/274644.274689. URL: http://dx.doi.org/10.1145/274644.274689.

Zhang, Li et al. (2008). "Spacetime faces: High-resolution capture for˜ modeling and animation". In: *Data-Driven 3D Facial Animation*. Springer, pp. 248–276.

Zhang, Peizhao et al. (2014). "Leveraging Depth Cameras and Wearable Pressure Sensors for Full-body Kinematics and Dynamics Capture". In: *ACM Trans. Graph.* 33.6, 221:1–221:14. ISSN: 0730-0301. DOI: 10.1145/2661229.2661286. URL: http://doi.acm.org/10.1145/2661229.2661286.

Zhou, Yang et al. (2018). "VisemeNet: Audio-Driven Animator-Centric Speech Animation". In: *arXiv preprint arXiv:1805.09488*.

Lists of Works

Peer Reviewed Journal

1. 加藤卓哉，深山覚，中野 倫靖，後藤真孝，森島繁生，“歌声と楽曲構造を入力とした歌唱時の表情アニメーション自動生成手法”，画像電子学会誌, April, 2019.
2. Pavel A. Savkin, 加藤卓哉，福里 司，森島繁生，〞老化時の皺の個人性を考慮した経年変化顔画像合成.〞情報処理学会論文誌, Vol.57, No.7, pp.1627-1637, July, 2016. 特選論文受賞
3. 福里 司，藤崎匡裕，加藤卓哉，森島繁生. 〞頭蓋骨形状を考慮した肥痩変化顔画像合成.〞画像電子学会誌, Vol.46, No.1, pp.197-205, February, 2017.

International Conference

Full Paper

1. Naoya Iwamoto*, Takuya Kato*, Hubert P. H. Shum, Ryo Kakitsuka, Kenta Hara, Shigeo Morishima (* equally contributed) “DanceDJ: A 3D Dance Animation Authoring System for Live Performance” Best Paper Award(Gold), Advances in Computer Entertainment Conference - ACE 2017, December 2017.
2. Takuya Kato, Shunsuke Saito, Masahide Kawai, Tomoyori Iwao, AkinobuMaejima, Shigeo Morishima, “Character Transfer: Example-based individuality retargeting for facial animation” (Acceptance rate 22%), International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision 2014 (WSCG2014), 121-129, June 2014

Short Paper / Techincal Brief

1. Pavel A. Savkin, Tsukasa Fukusato, Takuya Kato, Shigeo Morishima, “Wrinkles Individuality Preserving Aged Texture Generation using Multiple Expression Images”, Computer Vision Theory and Applications (VISAPP) 2018, Madeira, Portugal, 2018.01.27-29.

2. Shintaro Yamamoto, Pavel Savkin, **Takuya Kato**, Shoichi Furukawa, Shigeo Morishima, "Facial Video Age Progression Considering Expression Change", Computer Graphics International 2017, Short Paper, Yokohama, Japan, 2017.06.27-6.30.
3. Fumiya Narita, Shunsuke Saito, **Takuya Kato**, Tsukasa Fukusato, and Shigeo Morishima, "Garment Transfer for Quadruped Characters", Eurographics 2016, Lisbon, 2016.05.09-13.
4. Shugo Yamaguchi, **Takuya Kato**, Tsukasa Fukusato, Chie Furusawa, Shigeo Morishima, "Region-Based Painting Style Transfer", ACM SIGGRAPH Asia2015, Kobe, 2015.11.02-05.

Poster

1. Kanami Yamagishi, **Takuya Kato**, Shintaro Yamamoto, Ayano Kaneda and Shigeo Morishima "How Makeup Experience Changes how we see Cosmetics?", ACM Symposium on Applied Perception, Vancouver, Canada, 2018. 08. 10-2018.08.11.
2. Kanami Yamagishi, Shintaro Yamamoto, **Takuya Kato**, Shigeo Morishima. "Cosmetic Features Extraction by a Single Image Makeup Decomposition", CVPR2018, Women in Computer Vision Workshop, Salt Lake City, 2018.06. 22.
3. Yugo Sato, **Takuya Kato**, Naoki Nozawa, Shigeo Morishima. "Perception of Drowsiness based on Correlation with Facial Image Features" ACM Symposium on Applied Perception 2016, Anaheim, 2016.07.22-23.
4. Tsukasa Nozawa, **Takuya Kato**, Pavel Savkin, Naoki Nozawa, Shigeo Morishima, "3D Facial Geometry Reconstruction using Patch Database", ACM SIGGRAPH 2016, Anaheim, 2016.07.24-28.
5. Shoichi Furukawa, **Takuya Kato**, Pavel Savkin, Shigeo Morishima, "Video Reshuffling: Automatic Video Dubbing without Prior Knowledge" ACM SIGGRAPH 2016, Anaheim, 2016.07.24-28. ACM Student Research Competition 1st Place.
6. Savkin Pavel, Daiki Kuwahara, Masahide Kawai, **Takuya Kato**, Shigeo Morishima, "Wrinkles Individuality Representing Aging Simulation", ACM SIGGRAPH Asia2015, Kobe, 2015.11.02-05.
7. **Takuya KATO**, Ryo SUZUKI, Naomi OKAMURA, Taro KANAI, Akira KATO, Yuko SHIRAI, "Musasabi: 2D/3D intuitive and detailed visualization system for the forest" ACM SIGGRAPH2015, Los Angeles, 2015.08.09-13.

8. Hiroki Kagiyama, Masahide KAWAI, Daiki KUWAHARA, **Takuya KATO**, Shigeo MORISHIMA , “Automatic Synthesis of Eye and Head Animation According to Duration and Point of Gaze” ACM SIGGRAPH2015, Los Angeles, 2015.08.09-13.
9. Fumiya NARITA, Shunsuke SAITO, **Takuya KATO**, Tsukasa FUKUSATO and Shigeo MORISHIMA, “Texture Preserving Garment Transfer” ACM SIGGRAPH2015, Los Angeles, 2015.08.09-13.
10. Shugo YAMAGUCHI, Chie FURUSAWA, **Takuya KATO**, Tsukasa FUKUSATO, Shigeo MORISHIMA, "BGMaker: Example-Based Anime Background Image Creation from a Photograph" ACM SIGGRAPH2015, Los Angeles, 2015.08.09-13. ACM Student Research Competition Finalist (3rd Place)
11. Fumiya Narita, Shunsuke Saito, **Takuya Kato**, Tsukasa Fukusato and Shigeo Morishima. "Pose-Independent Garment Transfer." ACM SIGGRAPH ASIA 2014, Shenzhen, 2014.12.03-06.
12. **Takuya Kato**, Shunsuke Saito, Masahide Kawai, Tomoyori Iwao, Akinobu Maejima and Shigeo Morishima. "Example-Based Blendshape Sculpting With Expression Individuality", ACM SIGGRAPH 2014, 7, Vancouver, 2014.08.10-14.

Japanese Conference [Peer Reviewed]

Oral Presentation

1. 岩本尚也，**加藤卓哉** ，原健太，柿塚亮，森島繁生 "DanceDJ: ライブパフォーマンスを実現する実時間ダンス生成システム" WISS2017, 山梨, 2017.12.6 - 8.
2. 古川翔一，**加藤卓哉**，サフキンパーベル，森島繁生．〞顔の発話動作と音声とを同期させた映像を生成する手法の提案．〞顔学フォーラム 2016, 東京, 2016.11.19-20.
3. **加藤卓哉**，森島繁生．〞キャラクタの個性的な表情特徴を反映した表情モデリング法の提案．〞フォーラム顔学 2015, 名古屋, 2015.09.12-13.
4. 山口周吾，**加藤卓哉**，古澤知英，福里司，森島繁生．" BG Maker ~アニメ背景画生成システムの提案とゲーム応用の可能性. " Computer Entertainment Developers Conference 2015 (CEDEC2015), 横浜, 2015.08.26-28.
5. 成田史弥，斉藤隼介，**加藤卓哉**，福里司，森島繁生．〞ポーズに依存しない 4 足キャラクタ間の衣装転写システムの提案." Visual Computing/GCAD 合同シンポジウム 2015, 9, 姫路, 2015.06.28-29.(採択率:42%).

6. サフキン・パーベル，川井正英，桑原大樹，加藤卓哉，森島繁生．〞皺の発生過程を考慮した経年変化顔画像合成." Visual Computing/GCAD 合同シンポジウム 2015, 37, 姫路, 2015.06.28-29.(採択率:42%).
7. 加藤卓哉．森島繁生 "Character Transfer キャラクタ固有の表情特徴を考慮した顔アニメーション生成手法." Computer Entertainment Developers Conference 2014 (CEDEC2014), 横浜, 2014.09.02-04.

Poster

1. 鍵山裕貴，加藤卓哉，森島繁生．"注視時間と注視位置が変化する際のキャラクタの頭部及び眼球運動の自動合成." Computer Entertainment Developers Conference 2015 (CEDEC2015), 横浜, 2015.08.26-28.

Japanese Conference [Without Review]

Oral Presentation

1. 山本晋太郎，サフキンパーベル，加藤卓哉，山口周悟，森島繁生．“表情変化を考慮した経年変化顔動画合成．” 情報処理学会コンピュータグラフィックスとビジュアル情報学研究会 第 166 回研究発表会，2017.3.
2. 山本晋太郎，サフキンパーベル，佐藤優伍，加藤卓哉，森島繁生．“笑顔動画データベースを用いた顔動画の経年変化．” 情報処理学会第 79 回全国大会，2017.3.
3. 古川翔一，加藤卓哉，サフキンパーベル，森島繁生．〞フレームリシャッフリングに基づく事前知識を用いない吹替映像の生成．〞情報処理学会 第 78 回全国大会，横浜，2016.3.10-12.（口頭発表，査読なし） 学生奨励賞受賞，大会奨励賞受賞
4. 古川 翔一，加藤 卓哉，野澤 直樹，サフキン パーベル（早大），森島繁生（早大/JST CREST). "主成分分析に基づく類似口形状検出によるビデオ翻訳動画の生成." 情報処理学会 グラフィックスと CAD 研究会第 161 回研究発表会， 15, 神戸大学, 2015.11.06. Transactions on Computer Vision and Applications（CVA）推薦論文， 優秀研究発表賞

5. 野沢 綸佐，**加藤 卓哉**，藤崎 国裕，サフキン パーベル（早大），森島繁生（早大/JST CREST). "パッチタイリングを用いた法線推定による 3 次元顔形状復元.〞情報処理学会 第 199 回 CVIM 合同研究会, 15, 神戸大学, 2015.11.07. Transactions on Computer Vision and Applications（CVA）推薦論文
6. 成田史弥，斉藤隼介，**加藤卓哉**，福里司，森島繁生. "フィッティングを保持した体型の変化に頑健な衣装転写システムの提案." 情報処理学会 第 77 回全国大会，4Y-02, 京都大学, 2015.03.18. 学生奨励賞
7. サフキン・パーベル，川井正英，桑原大樹，**加藤卓哉**，森島繁生. "皺の個人性を考慮した経年変化顔画像合成" 情報処理学会 第 77 回全国大会, 4Y-05, 京都大学，2015.03.18.
8. サフキン・パーベル，川井正英，桑原大樹，**加藤卓哉**，森島繁生. "キャラクタ特有の特徴再現を考慮したリアルな表情リターゲッ ティング手法の提案." 情報処理学会 グラフィックスと CAD 研究会第 158 回研究発表会, 15, 理化学研究所, 2015.02.27. 優秀研究発表賞
9. **加藤卓哉**，斉藤隼介，川井正英，岩尾知頼，前島謙宣 森島繁生. "キャラクタに固有な表情変化の特徴を反映したキーシェイプ自動生成手法の提案." 情報処理学会 第 76 回全国大会, 4ZC-3, 東京電機大学, 2014.03.12. 学生奨励賞
10. **加藤卓哉**，川井正英，斉藤隼介，岩尾知頼，前島謙宣 森島繁生. "キャラクタ特有の特徴再現を考慮したリアルな表情リターゲッティング手法の提案." 情報処理学会 グラフィックスと CAD 研究会第 154 回研究発表会, 15, 理化学研究所, 2014.02.21.

Poster / Demo

1. 岩本尚也，**加藤卓哉** ，原健太，柿塚亮，森島繁生 "Dance DJ: ライブパフォーマンスのためのダンス動作ミックスシステム" WISS2016, 滋賀, 2016.12.14 - 17.
2. Shugo Yamaguchi, **Takuya Kato**, Tsukasa Fukusato, Chie Furusawa, Shigeo Morishima (Waseda Univ.). "Region-Based Painting Style Transfer." MIRU2015, SS5-33, 大阪, 2015.07.27 - 30.
3. **加藤卓哉**，斉藤隼介，川井正英，岩尾知頼，前島謙宣 森島繁生. "キャラクタ固有の表情特徴を考慮した顔アニメーション生成手法." Visual Computing/GCAD 合同シンポジウム 2014, 37, 東京, 2014.06.29-30.

謝辞

博士学位論文を執筆するにあたり，多くの方々のご指導とご助力を いただきました．
今日に至るまで終始ご指導賜り，素晴らしい研究の機会を与えてくださいました森島繁生教授に厚く御礼申し上げます． 高校生の頃，森島先生の研究を見て思い描いた「この先生の下で博士号を取りたい」という夢を，様々な形でご支援頂きましたことを心より感謝申し上げます．決して，優秀と言える学生でなかった私に，ありとあらゆる機会をご提供頂いたことは，私にとって何にも替え難い貴重な財産となりました． 心より感謝申し上げます．

本論文を提出するにあたり，3 名の先生に副査をお願いしてご助言を賜りました．小松進一教授には，大変お忙しい中様々な無理なお願いをしたにも関わらず，快く受け入れて頂きましたこと，心より感謝申し上げます．澤田秀之教授には，専門外の分野にも関わらず，様々なご助言を頂きましたこと，心より感謝申し上げます．国立研究開発法人産業技術総合研究所の深山覚様には，産総研のインターンをした時から，3 年以上様々な形でご支援，ご助言頂きました．特に歌唱動作合成の研究については，深山さん抜きには進めることはできず，そして課程内の卒業も叶わなかったかもしれません．長きに渡りお世話になりました深山さんを副査にお迎えできましたことを，心より幸せに思います．心より御礼申し上げます．

修士課程，博士課程で研究を進めるに辺り，学外の様々な方々に多大なる支援を頂きました．OLM Digital の安生健一氏，木村歩氏，国立研究開発法人産業技術総合研究所の後藤真孝氏，濱崎雅弘氏，中野倫靖氏，加藤淳氏，佃洸摂氏，小山裕己氏，渡邉研斗氏，土田修平氏，Paul Haimes 氏，佐藤めぐみ氏，笠井志麻氏，Walt Disney Company Japan の佐瀬正氏，堀切伸行氏，UCLA の Demetri Terzopoulos 氏，仲田真輝氏，UC Berkley の Alexei Efros 氏，Jun-Yan Zhu 氏，Taesung Park 氏，Philip Isola 氏，Northumbria University の Hubert P. H. Shum 氏に感謝申し上げます．また，それぞれのインターン先や留学先などで私に多大なる影響を与えてくださいました同期の方々に感謝申し上げます．皆様には，研究を進める上でのアドバイスはもちろん，研究のノウハウや心得など，今の私を構築する様々なことをご教示頂きました．私の研究がなかなか進まない時や，精神的に追い込まれている時，皆様のお言葉やご助言に助けられていたように思います．心より御礼申し上げます．

森島研究室で研究をするにあたり，森島研の博士 OB の先輩方には大変お世話になりました．OLM Digital の前島謙宣氏には，まだ右も左もわからない私の研究相談や人生相談に何度も何度も乗って頂き，たくさんの助言を頂きました．今思い返せば本当に無礼な態度な学生だった私に，研究者としての正しいあり方を示して頂きました．奈良先端科学技術大学院

大学の久保尋之氏には，研究などはもちろん SIGGRAPH の学生ボランティアの活動をご紹介いただき，私の人生に大きな影響を与えてくださいました．お二人のような森島研をずっと大切に思っていただける先輩方があっての森島研です．これから，私もお二人のように森島研を支える人間になりたいと思います．心より御礼申し上げます．

森島研で，私が博士在学中に博士課程で研究されておりました先輩方には，多大なる影響を受け，常に私の目標でありました．駒澤大学の平井辰典氏には，研究室にいらっしゃる頃はもちろん，駒澤大学に移られてからも，大変お世話になりました．厳しくも思いやりのある平井さんのお言葉は，いつまでも心に残って離れません．平井さんの叱咤激励は，私の博士課程の本当に重要な心の支えであったと常に感じております．尊敬する先輩として，これからも何卒よろしくお願い申し上げます．Huawei Japan の岩本尚也氏には，DanceDJ の研究で大変お世話になりました．岩本さんのご尽力でシステムが完成し，学会発表するに至るまで，様々な面でお世話になりました．私生活でもとても仲良くさせていただき，様々な影響を受けました．心より御礼申し上げます．東京大学の福里司氏には，最も身近な博士課程の先輩として，様々なことを教えていただきました．研究のことや研究室のことで，意見が度々対立することもありましたが，今思えば，様々な議論の中で得られたことも多くありました．心より御礼申し上げます．

森島研の研究員として，お世話になりました谷田川達也氏には，博士課程に入ってからの 3 年間で大変お世話になりました．谷田川さんの研究者としての姿勢，人間としての素晴らしさに多大なる影響を受け，私の博士課程の大きな心の支えとなりました．森島研の様々なプロジェクトに谷田川さんに関わっていただいたことは，私のみならず森島研の財産です．心より御礼申し上げます．

博士課程に進学するにあたり，リーディング大学院プログラム『実体情報学博士プログラム』の皆様に大変お世話になりました．特に，一期生として，研究生活のみならず私生活でも心の支えとなってくれた岡村尚美氏，加藤陽氏，金井太郎氏，古志知也氏，佐々木崇史氏，鈴木遼氏，津村遼介氏，山田竜郎氏，トモ　ティト・プラドノ氏に感謝申し上げます．研究室内に博士の同期がいない私にとって，彼らの影響はとても大きく，彼らと切磋琢磨して研究生活を送ることができましたことを心より幸せに思います．これからはそれぞれ異なる道を歩むことになりますが，私の人生の友として，共に歩んでくれることと思います．これからもよろしくお願いします．
また，こうした素晴らしい同期と出会い，様々な学びの機会をご提供くださいましたリーディング大学院の担当教員の皆様，事務局の皆様には，多大なる心配やご迷惑をおかけいたしました．皆様の多大なるご尽力なしにはここまで至ることはできませんでした．重ねて御礼申し上げます．

私の研究生活において，共に切磋琢磨して頑張ってきた森島研究室の皆様の方々に感謝申し上げます．中でも，直属の先輩として私を支えてくださいました，岩尾知頼氏，中村太郎氏，斉藤隼介氏，川井正英氏，桑原大樹氏，溝川あい氏には，大変お世話になりました．皆様のご支援あってこそ今の私であると強く感じております．心より御礼申し上げます．同期として切磋琢磨した岡本翠氏，後藤岳人氏，河村俊哉氏，藤貴大氏，藤崎匡裕氏，古澤知英氏と切磋琢磨して研究生活を送ることができましたことを心より幸せに思います．心より御礼申し上げます．山口周吾氏，野澤直樹氏，野沢綸佐氏，中塚貴之氏，金田綾乃氏，福原吉博氏には，博士課程の後輩として共に切磋琢磨できましたことを，心より幸せに思います．皆様の博士号取得を心より応援しております．また，同じ班の後輩として共に研究生活を送ってきた，鍵山裕貴氏，サフキン・パーベル氏，成田史弥氏，古川翔一氏，佐藤優伍氏，山本晋太郎氏，山岸奏実氏，夏目亮太氏，土屋志高氏，矢代達希氏に感謝申し上げます．これからも，皆様のご活躍を心よりお祈りしております．

最後に，ここまで経済的，精神的に私を支えてくださいました，父，母，姉，祖父，祖母，親戚の皆様，そして，私を明るく励ましてくれた妻に心より感謝申し上げます．

2019 年 2 月