Highly efficient facial blendshape animation with analytical dynamic deformations

X. Y. You* • F. Tian* • W. Tang*

Abstract Adding physics to facial blendshape animation is an active research topic. Existing physics-based approaches of facial blendshape animation are numerical, so they require special knowledge and skills, additional preprocess, large computer capacity, and expensive calculations leading to low animation frame rates, and are not easy to learn, implement and use. To tackle these problems, we propose an analytical approach and develop a blending force-based framework for physics-based facial animation. The proposed approach introduces the equation of motion to consider inertial effects, damping effects and the resistance against deformations, combines them with source and target facial shapes to formulate the mathematical model of dynamic deformations, and develops a simple and efficient closed-form solution. The blending force-based framework incorporates the new proposed slider force-based, exponentiation force-based and random force-based methods built on the obtained closed form solution to achieve highly efficient facial animation.

Compared with facial blendshape animation using geometric linear interpolation, the proposed approach is physics-based. It not only creates all the blended shapes generated by linear interpolation, but also a much larger superset of blended shapes. Unlike linear interpolation which can only generate blended shapes with a same deformation rate, the proposed approach can generate blended shapes with different deformation rates, resulting in special effects of acceleration and deceleration. Compared to existing physics-based approaches of facial blendshape animation which are numerical, the proposed approach is the first time to develop an analytical approach of physics-based facial blendshapes. It does not require any special knowledge and skills and is easy to learn, implement and use. More importantly, it can avoid the additional preprocess of numerical methods and create various physics-based facial blendshape animations highly efficiently. Moreover, it can be used to estimate physical parameters from real shapes and developed into an interactive and real-time physics-based shape manipulation tool.

Keywords facial blendshapes • physics-based facial animation • dynamic deformations • equation of motion • analytical solutions

1 Introduction

Facial blendshapes are the predominant choice for realistic facial animation in the movie industry and a standard feature of commercial animation packages [1]. They have driven animated characters in Hollywood films and attracted a lot of research attentions.

Facial blendshapes can be divided into geometric and physics-based. Linear interpolation plays a dominant role in geometric facial blendshapes. It can be mathematically formulated as [1]

$$\mathbf{x} = \mathbf{x}_0 + \sum_{j=1}^{N} w_j (\mathbf{x}_j - \mathbf{x}_0) \tag{1}$$

where $\mathbf{x}$, $\mathbf{x}_0$ and $\mathbf{x}_j$ contains the coordinates of all the vertices of a new facial shape (called the blended shape in this paper) to be created, a neutral shape, and the j^{th} blendshape (called the target shape), respectively, $0 \le w_j \le 1$ is a weight, and N is the total number of all the target shapes.

Linear interpolation based facial blendshapes are very popular since they have the advantages of simplicity, expressiveness and interpretability. In spite of this, the following limitations have been identified in [2]. First, the linear interpolation constrains the space of facial expressions to lie in an affine subspace. Since not all values of the weight w_j yield plausible facial deformations such as nonlinear and rotational deformations [3], the space of facial expressions should not be considered affine. Some approaches such as pose space deformation-based corrections [4] have been proposed to deal with nonlinearities which are not represented by the affine model. Second, facial blendshapes could be regarded as samples from a hypothesized manifold of facial

*Department of Creative Technology, Faculty of Science and Technology, Bournemouth University, Poole, BH12 5BB, United Kingdom

expressions. Generating a new facial shape requires enough target facial shapes to sample the manifold and define local linear interpolation functions. Creating enough target facial shapes is usually an iterative and labour intensive process. New techniques of facial blendshapes are required to create a superset of blended shapes which contain not only the subspace of the blended shapes generated by linear interpolation but also those which linear interpolation is unable to generate. Some efforts have been devoted to extend the subspace such as splitting the face model into segments and applying a local affine model to each segment [5] and capturing geometric shapes regulated by physics-inspired deformations [6, 7]. However, these approaches are difficult to consider complex physical behaviours. Third, linearly combining blendshapes tend to move groups of vertices jointly as blocks over time which cannot render fine temporal behaviours. Linear interpolation changes facial shapes with a same deformation rate. Generating special effects such as acceleration and deceleration effects [28] require animating facial shape changes with different deformation rates.

Physics-based facial blendshapes are to add physics to facial blendshapes which has a potential to tackle the above problems. Especially, when physics-based simulations are combined with data-driven approaches, realistic facial animation can be created. Apart from its capacity in tackling the above problems, another important feature of physics-based approaches is that the simulation parameters can be controlled to achieve the desired effects [8]. These simulation parameters include mass, damping coefficient, stiffness coefficient, rest shape volume, and static bone structure etc.

Existing physics-based facial animation is obtained by various numerical methods such as the finite element method which simulates facial models as thin shell [2] or a solid volume [8]. As stated in [3], "these methods can simulate all the desired dynamic effects, but are extremely difficult and time-consuming to set up" since they require special knowledge and skills, additional mesh preprocessing, large computer capacity, and high computational costs and are not easy to learn, implement and use. Therefore, they are not applicable to the situations where real-time animation or high animation frame rates are required. In facial animation, computational complexity is a fundamental constraint, and real-time performance is often much more important than a highly accurate facial shape [9].

In this paper, we will make the first attempt to develop analytical physics-based facial blendshape animation. Considering the previous work [2, 3] on facial animation uses Newton's second law of motion which does not take damping effects into account, we will use the equation of motion adopted in [10, 22] to incorporate inertial effects, damping effects, and the resistance against shape deformations. The mathematical model of dynamic deformations is obtained by combining the equation of motion with the deformation constraints from source and target facial shapes. An analytical solution of the mathematical model is derived and used to develop an efficient blending force-based animation framework consisting of slider force-based, exponentiation force-based and random force-based facial blendshapes which can create various blended facial shapes highly efficiently. Unlike existing numerical methods which must process polygon models first such as converting polygon meshes into finite element or mass-spring meshes, the proposed analytical approach directly uses polygon vertices for physics-based facial blendshapes.

The main technical contributions made by this paper are: 1) the mathematical model of dynamic deformations which integrates the equation of motion and the constraints of the source and target shapes; 2) the first analytical approach of dynamic facial blendshape animation which is easy to learn, implement and use by animators without special knowledge and skills of numerical calculations and able to create facial blendshape animation with high animation frame rates; 3) an easy-to-use and efficient blending force-based facial animation framework which integrates slider force-based, exponentiation force-based, and random force-based facial blendshapes.

The remaining parts of this paper are organized as follows. Section 2 briefly reviews the existing work followed by formulation of the mathematical model and analytical solutions of the mathematical model in Section 3. The blending force-based animation framework is investigated in Section 4. And the conclusions and further work are discussed in Section 5.

2. Related work

The proposed approach is related to facial shape interpolation and physics-based facial animation. In this section, we briefly review the research in these two fields.

Facial shape interpolation is the most intuitive and commonly used technique in facial animation practice. It can be divided into linear and nonlinear interpolation. Linear interpolation [11-15] plays a dominant role in facial blendshapes. Region-based linear interpolation of 3D face models proposed in [16] increases flexibility for modelling local deformations while keeping the model coherent. A bilinear model was applied to natural spatiotemporal phenomena such as dynamic faces and bodies [17]. A higher order generalisation of linear model called multilinear was used for modelling identity, expression, and speech independently in [18]. The cosine interpolation proposed in [19] and other variations such as spline can provide acceleration and deceleration

effects at the beginning and end of an animation. An optimization scheme was proposed in [20] to automatically explore the nonlinear relationship of facial blendshape animation from captured facial expressions.

Unlike these purely geometric face shape interpolation methods which blend source and target shapes through one weight or a set of weights, our proposed approach introduces one source shape and one target shape as the constraints of the equation of motion in the physics-based mathematical model, and uses blended forces and different values of the time variable involved in the equation of motion to create different blended shapes.

Physics-based facial animation is introduced to consider underlying physics of facial animation and tackle the problems of purely geometric facial animation. Various physics-based approaches have been proposed. Due to the limitation of space, here we only review mass-spring systems and the finite element method.

Mass-spring systems actuated by vector muscles were introduced to animate facial expressions in [21]. Video-based tracking of facial features was used to calculate muscle actuation parameters in [22]. Physically-plausible shape blending was obtained by linearly interpolating spring rest length parameters of a mass-spring system between source and target shapes [23].

The finite element method is the most popular numerical method especially in computational sciences and engineering. It was introduced to simulate facial surgery in [9]. 3D finite element simulations were integrated into a computer-aided surgical planning system in [24], and used to develop an anatomically accurate model for facial animation in [15]. Rig-space physics uses the underlying finite element model and explores the interactions between physical models and artistic rigging [25]. Physical face cloning was obtained through nonlinear finite element simulations with a neo-Hookean material [26]. Recently, finite element simulations were combined with facial blendshapes to preserve volume and avoid self-collisions during facial animation [2], used to enhance facial blendshape rigs [3], integrated with a data-driven approach to develop data-driven physics [27], and applied with a novel muscle activation model to achieve physics-based face modelling and animation [8].

Different from physics-based facial blendshapes using numerical methods which require an additional preprocess to convert polygon meshes into finite element or mass-spring meshes, special knowledge and skills to implement and use them, and large computer memory and high computational costs to realize the simulations leading to low animation efficiency, the proposed analytical approach directly works on the vertices of polygon models and avoids the problems of numerical methods.

3. Mathematical model and analytical solution

We use the same equation of motion adopted in [10, 22] to approximate facial deformations

$$m\frac{d^2x^{(i)}}{dt^2}+c\frac{dx^{(i)}}{dt}+kx^{(i)}=f^{(i)} \quad (i=1,2,3) \tag{2}$$

where m denotes mass which is used to provide inertia effects, c stands for the damping coefficient which is introduced to provide damping effects, k means the stiffness coefficient which reflects the resistance of facial skin against facial deformations, t is the time variable, the superscript i indicates the i^{th} component, i. e., $x^{(1)}=x$, $x^{(2)}=y$, and $x^{(3)}=z$ which represent the three components of deformations, and $f^{(1)}=f_x$, $f^{(2)}=f_y$, and $f^{(3)}=f_z$ indicate the three components of forces.

The face model at the neutral pose $t=0$ has no deformations, and the deformation rate at this pose is also zero. At the deformed pose $t=1$, the face model is deformed into various target shapes. Therefore, the following constraints can be obtained.

$$\begin{aligned} &t=0 \quad x^{(i)}=0 \quad \frac{dx^{(i)}}{dt}=0 \\ &t=1 \quad x^{(i)}=x_2^{(i)}-x_1^{(i)} \\ &(i=1,2,3) \end{aligned} \tag{3}$$

where the subscripts “1” and “2” indicate the neutral pose and deformed pose, respectively, and $x_1^{(i)}$ and $x_2^{(i)}$ stand for the coordinate values of the undeformed and deformed face models respectively.

Equations (2) and (3) are the mathematical model of dynamic facial blendshapes. Various numerical methods can be used to solve the mathematical model. However, these numerical methods have the problems discussed before. In this paper, we will develop a simple and efficient analytical solution of the mathematical model.

According to the theory of differential equations, the solution to nonhomogeneous second-order ordinary differential Equation (2) contains a general solution $x_h^{(i)}$ to the homogenous version of Equation (2) and a

particular solution $x_p^{(i)}$ caused by the right-hand-side mathematical expression. The general solution to the homogenous version of Equation (2) can be converted into a characteristic equation by letting

$$x_h^{(i)} = e^{rt} \quad (i = 1,2,3) \tag{4}$$

Substituting Equation (4) into (2), the homogenous version of the nonhomogeneous second-order ordinary differential equation (2) are changed into the following characteristic equation

$$mr^2 + cr + k = 0 \tag{5}$$

The two roots of Equation (5) can be written into the following form

$$r_{1,2} = \frac{-c \pm \sqrt{c^2 - 4mk}}{2m} \tag{6}$$

Depending on the different values of $c^2 - 4mk$, the roots given by Equation (6) and the analytical solution to Equations (2) and (3) can be divided into the following three cases to be discussed in Subsection 3.1, 3.2, and 3.3, respectively.

3.1 Analytical solution for $c^2 - 4mk = 0$

When $c^2 - 4mk = 0$, Equation (6) gives two repeated real roots below

$$r_{1,2} = -\frac{c}{2m} = q \tag{7}$$

The general solution of the homogenous version of Equation (2) for this case is

$$x_h^{(i)} = (c_1 + c_2 t)e^{qt} \quad (i = 1,2,3) \tag{8}$$

where c_1 and c_2 are two integration constants.

The particular solution of the nonhomogeneous second-order ordinary differential equation (2) can be taken to be

$$x_p^{(i)} = c_3 + c_4 t \quad (i = 1,2,3) \tag{9}$$

where c_3 and c_4 are two unknown constants.

Substituting Equation (9) into (2), the two unknown constants are determined. Introducing them back into Equation (9), the particular solution is obtained below.

$$x_p^{(i)} = f^{(i)} \quad (i = 1,2,3) \tag{10}$$

Putting Equations (8) and (10) together, the solution to Equation (2) is found to be

$$x^{(i)} = x_h^{(i)} + x_p^{(i)} = (c_1 + c_2 t)e^{qt} + f^{(i)} \quad (i = 1,2,3) \tag{11}$$

The two integration constants c_1 and c_2 can be determined by substituting Equation (11) into the constraints (3) which gives

$$x^{(i)} = \frac{1 + (-1 + tq)e^{qt}}{k} f^{(i)} \quad (i = 1,2,3) \tag{12}$$

where

$$f^{(i)} = \frac{k\left[x_2^{(i)} - x_1^{(i)}\right]}{1 + (-1 + q)e^{q}} \quad (i = 1,2,3) \tag{13}$$

3.2 Analytical solution for $c^2 - 4mk > 0$

When $c^2 - 4mk > 0$, Equation (6) gives two different real roots below

$$r_1 = \frac{-c + \sqrt{c^2 - 4mk}}{2m} = q_1$$
$$r_2 = \frac{-c - \sqrt{c^2 - 4mk}}{2m} = q_2 \tag{14}$$

The general solution of the homogenous version of Equation (2) for this case is

$$x_h^{(i)} = c_1 e^{q_1 t} + c_2 e^{q_2 t} \quad (i = 1,2,3) \tag{15}$$

Putting the above equation and Equation (10) together, the solution to Equation (2) is

$$x^{(i)} = c_1 e^{q_1 t} + c_2 e^{q_2 t} + f^{(i)} \quad (i = 1,2,3) \tag{16}$$

Substituting Equation (16) into the constraints (3) to determine the two integration constants c_1 and c_2, and introducing them back into Equation (16), we obtain

$$x^{(i)} = \frac{q_1 - q_2 + q_2 e^{q_1 t} - q_1 e^{q_2 t}}{k(q_1 - q_2)} f^{(i)} \quad (i = 1,2,3) \tag{17}$$

where

$$f^{(i)} = \frac{k(q_1 - q_2)\left[x_2^{(i)} - x_1^{(i)}\right]}{q_1 - q_2 + q_2 e^{q_1} - q_1 e^{q_2}} \quad (i = 1,2,3) \tag{18}$$

3.3 Analytical solution for $c^2 - 4mk < 0$

When $c^2 - 4mk < 0$, Equation (6) gives two different complex roots below

$$r_{1,2} = q_3 \pm j q_4 \tag{19}$$

where j is an imagery number, and

$$q_3 = -\frac{c}{2m} \qquad q_4 = \frac{\sqrt{4mk - c^2}}{2m} \tag{20}$$

The general solution of the homogenous version of Equation (2) for this case is

$$x_h^{(i)} = e^{q_3 t}\left(c_1 \cos q_4 t + c_2 \sin q_4 t\right) \quad (i = 1,2,3) \tag{21}$$

Putting the above equation and Equation (10) together, we obtain the solution to Equation (2)

$$x^{(i)} = e^{q_3 t}\left(c_1 \cos q_4 t + c_2 \sin q_4 t\right) + f^{(i)} \quad (i = 1,2,3) \tag{22}$$

Substituting Equation (22) into the constraints (3) to determine the two integration constants c_1 and c_2, and introducing them back into Equation (22), we obtain

$$x^{(i)} = \frac{q_4 + e^{q_3 t}\left(q_3 \sin q_4 t - q_4 \cos q_4 t\right)}{k q_4} f^{(i)} \quad (i = 1,2,3) \tag{23}$$

where

$$f^{(i)} = \frac{k q_4 \left[x_2^{(i)} - x_1^{(i)}\right]}{e^{q_3}\left(q_4 e^{-q_3} + q_3 \sin q_4 - q_4 \cos q_4\right)} \quad (i = 1,2,3) \tag{24}$$

From one of Equations (13), (18) and (24), we can obtain the force functions. Substituting them into the corresponding equations (12), (17) or (23), we can calculate coordinate values of the blended shapes at time t. In the following section, we will use them to develop an analytical blending force-based animation framework.

4. Blending force-based animation framework

Unlike existing geometric facial blendshapes which obtain new shapes by interpolating between one source shape and target shapes, our proposed approach will use the time variable t in Equations (12), (17) and (23) to interpolate the force $f^{(i)}$ which continually deforms one source shape into one or more target shapes. As shown in Equations (12-13), (17-18) and (23-24), $x_1^{(i)}$ presents a source shape, $x_2^{(i)}$ stands for a target shape, and $x^{(i)}$ is a blended shape. Equations (12), (17) and (23) are the functions of time variable t only. Therefore, we can implement a time slider to interpolate the force $f^{(i)}$ for each of the combinations between one source shape $x_1^{(i)}$ and each of target shapes $x_2^{(i)}$.

Existing facial blendshapes use weight sliders to directly blend among source and target shapes. In contrast, our proposed approach uses weight sliders to blend the forces obtained from one source shape and each of target shapes, and use the blended forces to create new blended shapes. Such a method of manipulating time and weight sliders to interpolate and blend forces and using the blended forces to create new blended shapes is called slider force-based facial blendshapes.

A professional blendshape model can involve 100 or more weights [1]. Manually manipulating such many weight sliders are time-consuming and can be difficult to find all or desired blended shapes. In order to avoid manually manipulate many sliders, we will propose exponentiation force-based and random force-based facial blendshapes to automatically generate blended shapes for animators to select from them.

The proposed blending force-based animation framework will integrate slider force-based, exponentiation force-based, and random force-based facial blendshapes which will be introduced in Subsection 4.1, 4.2 and 4.3 below, respectively. The generated facial blendshape animations are shown in the accompanied video.

4.1 Slider force-based facial blendshapes

Facial blendshapes can be divided into two different types. One is between one source shape and one target shape, and the other is between one source shape and two or more target shapes. The slider force-based method is applicable to both types of facial blendshapes

We have implemented Equations (12), (17) and (23) into time sliders to manipulate forces which continually deform the source shape into the corresponding target shape.

With the implemented slider force-based facial blendshapes, we first compare the blended shapes generated by the proposed approach and linear interpolation.

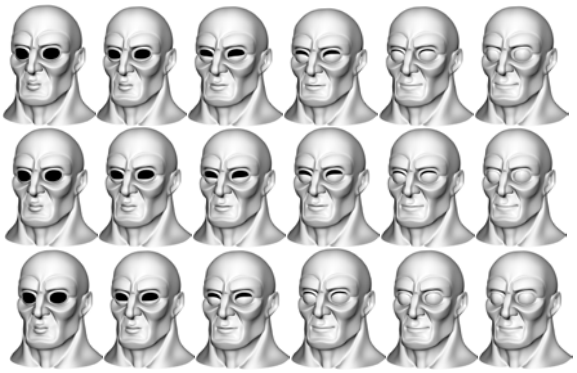

Fig. 1 Comparison of blended shapes generated by the proposed approach and linear interpolation

The parameters q and q_i ($i = 1,2,3,4$) involved in Equations (12-13), (17-18) and (23-24) depend on mass m, damping coefficient c, and stiffness coefficient k. It indicates that different mass, damping coefficient, and stiffness coefficient can be used to achieve different facial shapes through different values of $x^{(i)}$. Here we take two groups of different values to demonstrate this and compare with facial blendshapes using linear interpolation. The first group is: $m = 1.5$, $c = 8$, and $k = 10$, and the second group is: $m = 0.8$, $c = 7.5$, and $k = 20$. The source shape of a cartoon face at the neutral pose $t = 0$ and its target shape at the deformed pose $t = 1$ obtained from the link https://www.creativebloq.com/3d-world/download-files-3d-world-211-61420631 are shown in the column (a) and column (f) of Figure 1.

The blended shapes at the poses $t = 0.2$, 0.4, 0.6 and 0.8 from the first group of values are shown in (b)-(e) of the first row of Figure 1. The blended shapes from linear interpolation are shown in (b)-(e) of the second row. And those from the second group of values at the same poses are shown in (b)-(e) of the third row.

Comparing the images depicted in the figure, three conclusions can be drawn. First, the analytical approach of the dynamic facial blendshapes proposed in this paper uses the first group of mass, damping coefficient, and stiffness coefficient to achieve the same visual results (First row of Figure 1) of facial blendshapes as those generated by linear interpolation (Second row). It indicates that the proposed approach has a capacity to generate all the blended shapes generated by linear interpolation. Second, the analytical approach of the dynamic facial blendshapes uses the second group of mass, damping coefficient and stiffness coefficient to achieve different results as clearly demonstrated by the images shown in (c), (d) and (e) of the third row. It indicates that the blended shapes generated by linear interpolation are a subset of the blended shapes generated by the approach proposed in this paper. The proposed approach can generate a larger superset of blended shapes than linear interpolation. When two or more target shapes are blended together, the superset generated by the proposed approach will become much bigger than the subset generated by linear interpolation. Third, linear interpolation can only generate blended shapes with a same deformation rate. Differently, the proposed approach can generate blended shapes with different deformation rates, i. e., acceleration and deceleration effects stated in [28]. As shown in the figure, the shape changes from (b) to (d) of the third row generated by the proposed approach demonstrate a larger deformation rate than linear interpolation, and the shape changes from (d) to (f) of the third row generated by the proposed approach demonstrate a smaller deformation rate than linear interpolation. This feature is useful since it can be used to create special acceleration and deceleration effects [28].

Figure 2 shows more applications of the proposed approach in creating blended shapes between one source shape (first column of Figure 2) and seven target shapes (last column).

We have calculated the CPU time and found that obtaining all the new coordinate values of 14,232 polygon vertices for a facial animation of 1,389 frames (1,389 blended facial shapes) only takes one second on a PC with Intel® Xeon® CPU E5-1650 V2 @ 3.5 GHz and 32 GB of memory. The numerical method based on the physics-based model of soft tissues including muscle activation described in [8] takes more than two minutes on a laptop with a 3.1 GHz Intel Core i7 processor and 16 GB of main memory to generate a new facial shape for a model with 6,393 surface vertices and 8,098 volumetric vertices. Our proposed analytical approach of physics-based facial blendshapes is highly efficient.

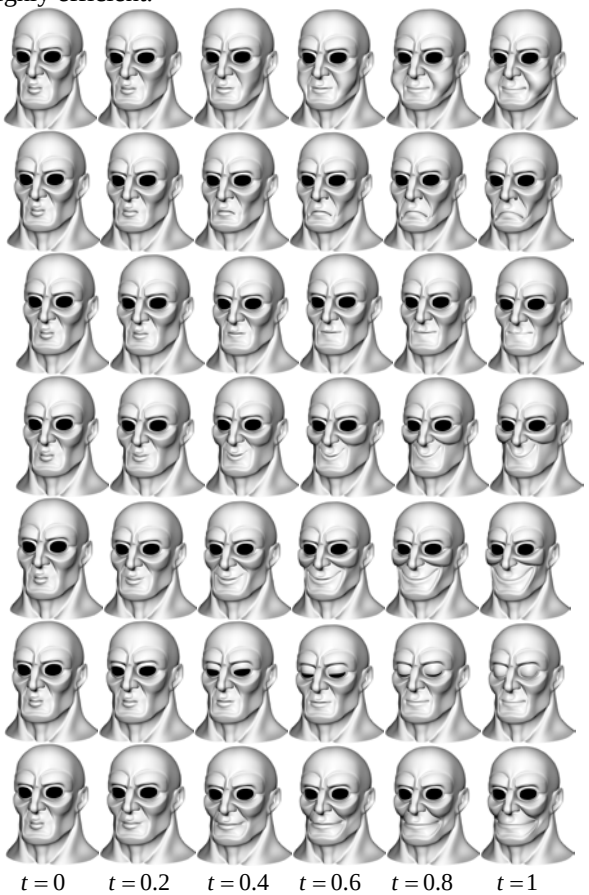

Fig. 2 Blended facial shapes of a cartoon face created by using slider force-based facial blendshapes

We have also used the source shape of a human face at the neutral pose $t=0$ and the target shapes at the deformed pose $t=1$ obtained from the link http://people.csail.mit.edu/sumner/research/deftransfer/data.html to create facial blendshapes. The obtained blended shapes at the poses $t=0.2$, 0.4, 0.6 and 0.8 are depicted in Figure 3 where the first, second and third rows show cry, fury and laugh expressions, respectively, and the first column is the source shape, and the last column is the target shapes. These images indicate the proposed approach successfully created new blended shapes from a source and a target shape.

$t=0$ $t=0.2$ $t=0.4$ $t=0.6$ $t=0.8$ $t=1$

Fig. 3 Blended facial shapes of a human face created by using slider force-based facial blendshapes

4.2 Exponentiation force-based facial blendshapes

In order avoid manually manipulate the implemented time sliders to generate new blended shapes for the situations with a lot of time sliders, this subsection and the following subsection will develop two methods to automatically generate blended shapes.

Assuming we have one source shape and N target shapes. By calculating the difference $x_2^{(i)}-x_1^{(i)}$ between each of the N target shapes $x_2^{(i)}$ and the source shape $x_1^{(i)}$, we obtain one force $f^{(i)}$ from Equations (13), (18) and (24). In total, N forces $f_j^{(i)}$ $(j=1,2,3,\cdots,N)$ are obtained. The blended force $f_b^{(i)}$ is obtained from the following linear combination of the N forces

$$f_b^{(i)}=\sum_1^N w_j f_j^{(i)} \qquad \sum_1^N w_j=1 \qquad (0\le w_j\le 1) \tag{25}$$

If each of the weight w_j ($j=1,2,3,...,N$) is uniformly discretized into L discrete values $w_{jl}=l/L$ (l=0, 1, 2, 3, …, L). The total weight combinations will be L^N.

For example, if we take the left one of Figure 4 to be a source shape, and the remaining 5 shapes in the same figure to be target shapes, we have $N=5$. If each of the weight w_j ($j=1,2,3,...,5$) is discretized into $L=3$ discrete values, we obtain total $3^5=243$ weight combinations.

Since $w_1=w_2=w_3=w_4=w_5=0$ leads to a zeroed blending force which will not generate any new shapes, the actual blended forces are 242 which can be used to create blended shapes at any poses in the time interval $0\le t\le 1$.

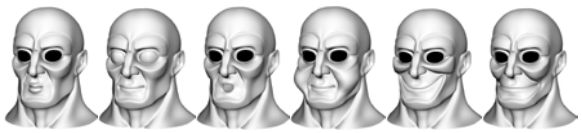

Fig. 4 Source shape and target shapes of a cartoon face used in exponentiation force-based facial blendshapes (Figure 5) and random force-based facial blendshapes (Figure 8)

At each time instant t, we can use the 242 blended forces to create 242 blended shapes. If we consider the time instants t=0, 0.2, 0.4, 0.6, 0.8, and 1, we obtain 1,452 blended shapes. Figure 5 gives the 48 blended shapes taken from every 5 ones of the created 242 blended shapes at $t=1$.

Having obtained the 1,452 blended shapes, we can view them and select some preferred ones from them.

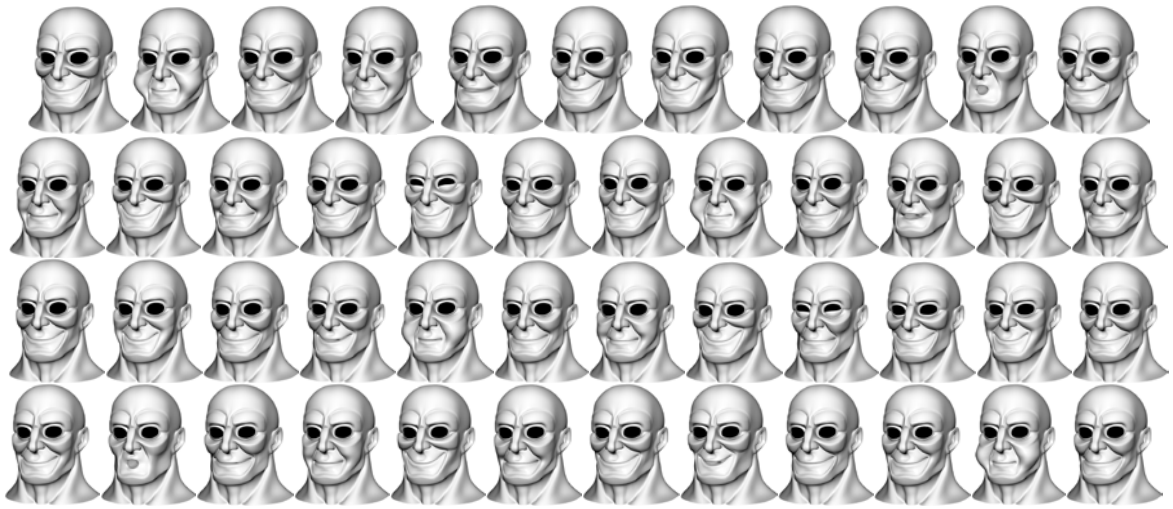

Fig. 5 Blended shapes of a cartoon face created by using exponentiation force-based facial blendshapes

For the human face shown in Figure 3, we use more expressions given in Figure 6 to demonstrate how to create new blended shapes with the approach of exponentiation force-based facial blendshapes. In Figure 6, the image from the left to the right shows the neutral, cry, fury, rage, surprise, and sad expressions, respectively.

Neutral Cry Fury Rage Surprise Sad

Fig. 6 Source shape and target shapes of a human face used in exponentiation force-based facial blendshapes (Figure 7) and random force-based facial blendshapes (Figure 9)

With the same method used to generate blended shapes shown in Figure 5, the 5 facial expressions shown in Figure 6 determine 242 blended forces at each time instant t which are used to create 242 blended shapes at the time instant. If we consider the time instants t =0, 0.2, 0.4, 0.6, 0.8, and 1, we obtain 1,452 blended shapes. Figure 7 gives 24 blended shapes taken from the 1,452 blended shapes.

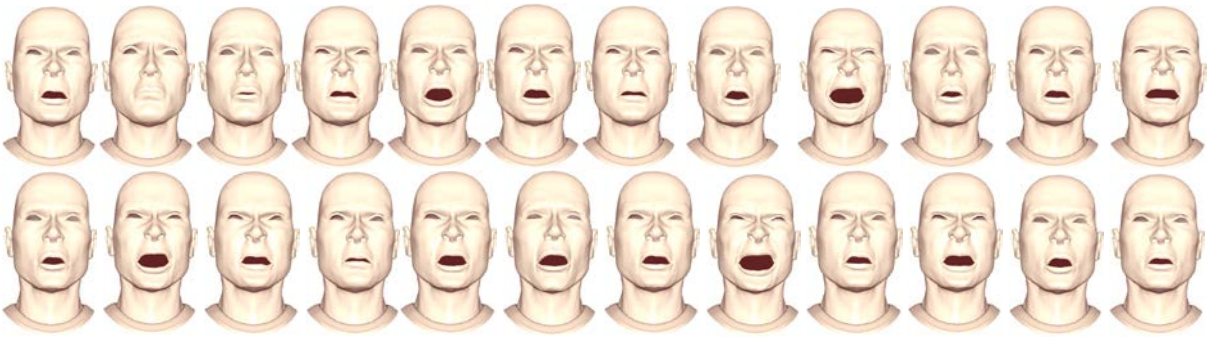

Fig. 7 Blended shapes of a human face created by using exponentiation force-based facial blendshapes

The above method of determining new blended forces and creating new blended shapes involves exponentiation. We call it exponentiation force-based facial blendshapes.

4.3 Random force-based facial blendshapes

Exponentiation force-based facial blendshapes give all L^N blended shapes. However, when N and L or one of them are very big, the exponentiation force-based facial blendshapes will generate a very large number of blended shapes. For example, when the discrete weights are increased from 0, 0.5 and 1 to 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 1, the blended forces are increased from 242 to 161,050. If we use these 161,050

blended forces to create new blended shapes at the time instants t =0, 0.2, 0.4, 0.6, 0.8, and 1, we obtain 966,300 blended shapes. Viewing all 966,300 blended shapes and selecting some from them is a time-consuming task. This problem can be overcome by generating uniformly distributed random numbers and using the generated random numbers to create new blended shapes for animators to view and select from them. In what follows, we will introduce this method.

If we have N target shapes, the weights will be w_1, w_2, w_3, ..., w_N according to Equation (25). All possible combinations of the N weights are all the permutations of 1, 2, 3,…, N respectively taken from the N numbers 1, 2, 3,…, N. The total number of all the permutations is $\overline{N} = \sum_{r=1}^{N} {}^{N}P_r$ where ${}^{N}P_r = \frac{N!}{(N-r)!}$. All the permutations are kept in a 2D array Per[n_1][n_2] ($1 \le n_1 \le \overline{N}; 1 \le n_2 \le N$). For example, when N=5, the total number of all the permutations will be $\overline{N} = 325$, the permutations of 1 taken from the 5 numbers 1,2,3,4,5 will be kept as Per[1][1-5]={1 0 0 0 0}, Per[2][1-5]={0 2 0 0 0},…, Per[5][1-5]={0 0 0 0 5}], the permutations of 2 taken from the 5 numbers will be kept as Per[6][1-5]={1 2 0 0 0},…, Per[25][1-5]={5 4 0 0 0}, and the last permutation of 5 taken from the total 5 numbers will be kept as Per[325][1-5]={5 4 3 2 1}.

From the source shape and the N target shapes, we can obtain N forces $f_j^{(i)}$ ($j = 1,2,3,...,N$). If we want to select L weight combinations from the total $\overline{N}$ weight combinations to create new blended shapes, we generate L uniformly distributed random numbers within the range between 1 and $\overline{N}$, and use the generated L random numbers to identify which weight combinations should be used.

Taking N=5 as an example, if the generated random numbers contain 2, 25, and 325, the weight combinations Per[2][1-5]={0 2 0 0 0}, Per[25][1-5]={5 4 0 0 0}, and Per[325][1-5]={5 4 3 2 1} are identified and converted into Per[2][1-5]={0 1 0 0 0}, Per[25][1-5]= {5/9 4/9 0 0 0} and Per[325][1-5]={5/15 4/15 3/15 2/15 1/15}.

Introducing each of the three weight combinations into Equation (25), the three blended forces are: $f_2^{(i)}$, $\left\lfloor 5f_1^{(i)} + 4f_2^{(i)} \right\rfloor / 9$, and $\left\lfloor 5f_1^{(i)} + 4f_2^{(i)} + 3f_3^{(i)} + 2f_4^{(i)} + f_5^{(i)} \right\rfloor / 15$ respectively. Substituting each of the three obtained blended forces into Equations (12), (17) and (23), the new blended shapes between $t = 0$ and $t = 1$ can be created.

If we take L = 50 and use the source (neutral) shape and the 5 target shapes shown in Figure 4 which gives N=5, we create 50 new blended shapes at each of t values from 50 blended forces. Figure 8 gives the first 48 ones of the 50 new blended shapes at the pose $t = 1$.

Since only 50 random numbers are used to create blended shapes, the blended shapes very close to $w_j = 1$ are not included in the 50 blended shapes. More random numbers can be used to improve this.

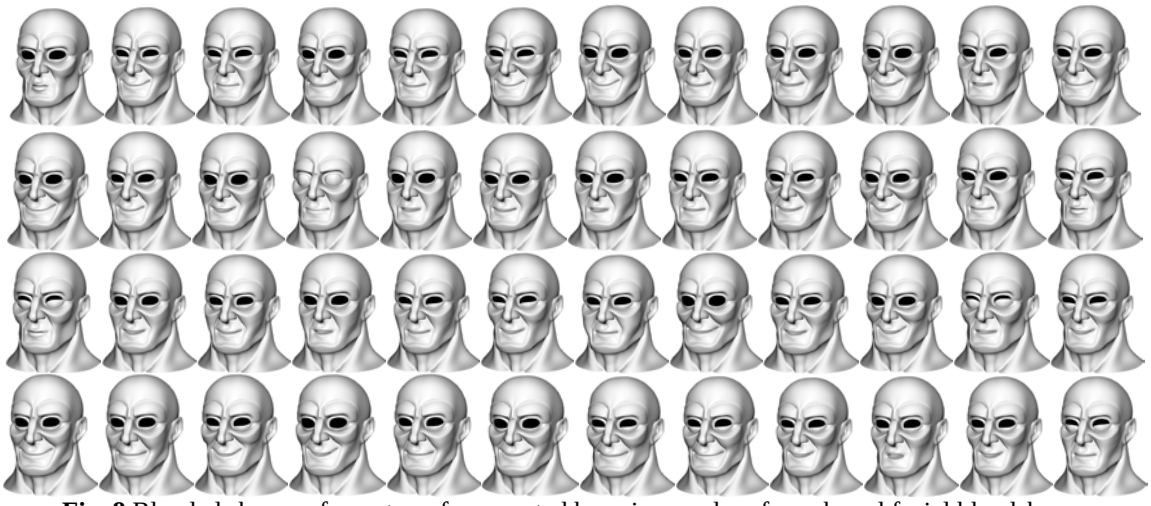

Fig. 8 Blended shapes of a cartoon face created by using random force-based facial blendshapes

Still taking L = 50 but using the source (neutral) shape and the 5 target shapes shown in Figure 6, we have N=5, and create 50 new blended shapes at each of t values from 50 blended forces. Figure 9 gives 24 blended shapes of the 50 new blended shapes at the pose $t = 0.6$.

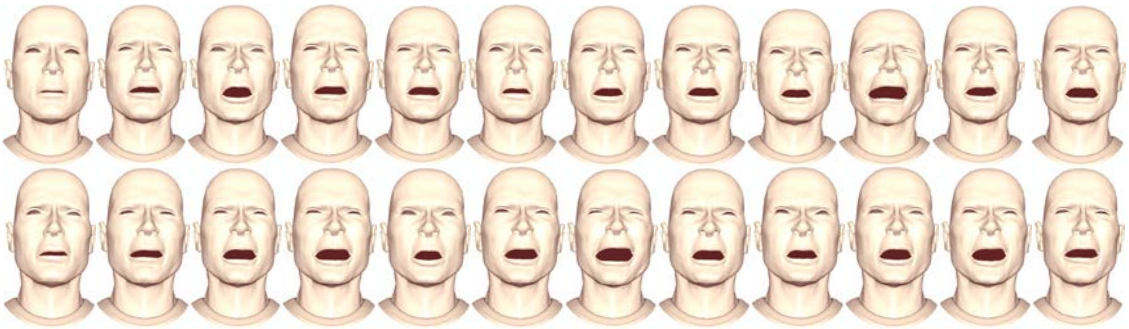

Fig. 9 Blended shapes of a human face created by using random force-based facial blendshapes

5. Conclusions and future work

In this paper, we have developed a novel approach of physics-based facial blendshapes. The approach is based on the analytical solution to the mathematical model integrating the equation of motion and the constraints of source and target facial shapes. It is used to develop an easy-to-use and efficient facial animation framework which integrates slider force-based, exponentiation force-based, and random force-based facial blendshapes.

The experiments made in this paper indicate that the proposed analytical physics-based approach and blending force-based animation framework can create a superset of blended shapes which not only contains the blended shapes generated by linear interpolation, but also those which cannot be generated by linear interpolation. They can create blended shapes with different deformation rates and overcome the limitation of linear interpolation which can only generate blended shapes with a same deformation rate. The analytical equations and their use in facial blendshapes demonstrate simplicity, easiness, low computer capacity requirement, and very high computational efficiency of creating facial animation.

The proposed approach offers a rich set of opportunities for future research. First, the influences of facial muscle, other facial tissues, and rigid bone structures on facial deformations can be introduced into the proposed approach. This can be achieved by transforming their influences into external forces to be included in the right-hand-side force functions. Second, the physical parameters, i. e., mass, damping coefficient and stiffness coefficient can be estimated to create realistic facial deformations. Different mass, damping coefficient and stiffness coefficient at different vertices [22] can be well estimated by minimizing the differences between the captured real facial shapes with the blended facial shapes created by the proposed approach. Third, the developed analytical solutions are used as a shape blending tool in this paper. Actually, they can be also used as a shape manipulation tool. This can be obtained by using the force functions as sculpting forces. The sculpting forces can be taken to be constants, linear variations or nonlinear variations and the corresponding particular solutions can be derived and added to the general solutions to develop powerful physics-based shape manipulation handles and create different deformation effects.

References

[1] Lewis JP, Anjyo K, Rhee T, Zhang M, Oighin F, Deng Z (2014) Practice and theory of blendshape facial models. In Eurographics 2014-State of the Art Reports, Eurographics Association, pp. 199-218

[2] Barrielle V, Stoiber N, Caqniart C (2016) Blendforces: A dynamic framework for facial animation. Computer Graphics Forum (Eurographics 2016) 35(2):341-352

[3] Kozlov Y, Bradley D, Bächer M, Thomaszewski B, Beeler T, Gross M (2017) Enhancing facial blendshape rigs with physical simulation. Computer Graphics Forum (EUROGRAPHICS 2017) 36(2):75-84

[4] Seol Y, Seo J, Kim PH, Lewis JP, Noh J (2012) Weighted pose space editing for facial animation. The Visual Computer 28(3):319-327

[5] Tena JR, De la Torre F, Matthews I (2011) Interactive region-based linear 3D face models. ACM Transactions on Graphics (SIGGRAPH 2011) 30(4):76

[6] Ma W-G, Wang Y-H, Fyffe G, Barbič J, Chen B-Y, Debevec P (2011) A blendshape model that incorporates physical interaction. Proceeding of SIGGRAPH Asia 2011 Posters, Article No. 35

[7] Li H, Yu J, Ye Y, Bregler C (2013) Realtime facial animation with on-the-fly correctives. ACM Transactions on Graphics (SIGGRAPH 2013) 32(4):42

[8] Ichim A-E, Kadleček P, Kavan L, Pauly M (2017) Phace: Physics-based face modelling and animation. ACM Transactions on Graphics (SIGGRAPH 2017) 36(4):153

[9] Koch RM, Gross MH, Carls FR, von Büren DF, Fankhauser G, Parich YIH (1996) Simulating facial surgery using finite element methods. Proceeding of SIGGRAPH 1996, pp. 421-428
[10] Park SI, Hodgins JK (2008) Data-driven modelling of skin and muscle deformation. ACM Transactions on Graphics (SIGGRAPH 2008) 27(3):96
[11] Bergeron P, Lachapelle P (1985) Controlling facial expressions and body movements in the computer generated animated short 'Tony de Peltrie'. In SIGGRAPH '85 Tutorial Notes, Advanced Computer Animation Course
[12] Pighin F, Hecker J, Lischinski D, Szeliski R, Salesin DH (1998) Synthesizing realistic facial expressions from photographs. Proceedings of SIGGRAPH 1998, pp. 75-84
[13] Seo H, Thalmann NM (2003) An automatic modeling of human bodies from sizing parameters. Proceedings of the 2003 Symposium on Interactive 3D Graphics, pp. 19-26
[14] Liang R-H, Ouhyoung M, (2004) Surface detail capturing for realistic facial animation. Journal of Computer Science and Technology 19(5):618-625
[15] Sifakis E, Neverov I, Fedkiw R (2005) Automatic determination of facial muscle activations from sparse motion capture marker data. ACM Transactions on Graphics (SIGGRAPH 2005) 24(3):417-425
[16] TENA JR, (2011) Interactive region-based linear 3D face models. ACM Transactions on Graphics (SIGGRAPH 2011) 30(4):72
[17] Akhter I, Simon T, Khan S, Matthews I, Sheikh Y (2012) Bilinear spatiotemporal basis models. ACM Transactions on Graphics 31(2):17
[18] Vlasic D, Brand M, Pfister H, Popović J (2005) Face transfer with multilinear models. ACM Transactions on Graphics (SIGGRAPH 2005) 24(3):426-433
[19] Waters K, Levergood TM (1993) Decface: An automatic lip-synchronization algorithm for synthetic faces. Cambridge Research Laboratory - Technical Report Series, pp. 1-25
[20] Liu X, Xia S, Fan Y, Wang Z (2011) Exploring non-linear relationship of blendshape facial animation. Computer Graphics Forum 30(6):1655-1666
[21] Platt SM and Badler NI (1981) Animating facial expressions. Proceedings of SIGGRAPH 1981, pp. 245-252
[22] Terzopoulos D, Waters K (1993) Analysis and synthesis of facial image sequences using physical and anatomical models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(6):569-579
[23] Ma W-C, Wang Y-H, Fyffe G, Chen B-Y, Debevec P (2012) A blendshape model that incorporates physical interaction. Computer Animation and Virtual Worlds 23(3-4):235-243
[24] Keeve E, Girod S, Pfeifle P, Girod B (1996) Anatomy-based tissue modeling using the finite element method. Proceedings of the 7th Conference on Visualization, pp. 21-27
[25] Hahn F, Martin S, Thomaszewski B, Sumner R, Coros S, Gross M (2012) Rig-space Physics. ACM Transactions on Graphics (SIGGRAPH 2012) 31(4):72
[26] Bickel B, Kaufmann P, Skouras M, Thomaszewski B, Bradley D, Beeler T, Jackson P, Marschner S, Matusik W, Gross M (2012) Physical face cloning. ACM Transactions on Graphics (SIGGRAPH 2012) 31(4):118
[27] Kim M, Pons-Moll G, Pujades S, Bang S, Kim J, Black MJ, Lee S-H (2017) Data-driven physics for human soft tissue animation. ACM Transactions on Graphics (SIGGRAPH 2017) 36(4):54
[28] Noh J-Y, Neumann U (1999) A survey of facial modeling and animation techniques. Tech. Rep. USC-TR-99-705, University of Southern California, pp. 1-26

Author biographies

Mr Xiangyu You complete his secondary education at Bournemouth School, obtained his BSc degree from University of Warwick, and MSc degree from Coventry, UK. He is currently working for Accenture, a company among Fortune Global 500 which focuses on consulting, technology services and outsourcing. He is currently researching into modelling and animation of virtual humans including facial blendshape animation.

Dr Feng Tian is an Associate Professor at the Department of Creative Technology, Faculty of Science and Technology, Bournemouth University. He has been researching in the areas of Computer Graphics, Computer Animation, NPR, Augmented Reality, Virtual Reality, etc. and has published over 90 papers in peer reviewed international journals and conferences. Prior to joining in Bournemouth University, he was an Assistant Professor in the School of Computer Engineering, Nanyang Technological University (NTU), Singapore. As a visiting scholar, he has been visiting and collaborating with a number of universities, including Paris University XI, France, New South-Wales University, Australia, Tokyo Institute of Technology, Japan, etc.

Dr Wen Tang is a full Professor of Games Technology in the Department of Creative Technology, Faculty of Science and Technology, Bournemouth University, the head of the Centre for Games and Music Technology Research, and the lead developer of the BU Games Analytics Platform. She was originally an Engineering graduate from China, and made transition into Computer Science during her PhD study in Computer Graphics and Computer-Aided Design at the University of Leeds, UK. Before joining Bournemouth University, she was a research fellow at Universities of Leeds and Bradford on two EU funded projects on collaborative virtual reality software technologies. After that, she was Senior Lecturer, then Reader in Computer Graphics and Games Technology in School of Computing at Teesside University. Apart from publishing many papers in academic journals and international conferences, she is an editor of 5 conference proceedings, and principal investigator and co-investigator of a large number of external funded projects funded by EPSRC, Innovate UK, H2020, and Royal Academy of Engineering etc.

Author photographs

Mr Xiangyu You

Associate Prof Feng Tian

Prof Wen Tang

Click here to access/download

Video Clip

Facial blendshapes.mp4