10-2010

Modeling 3D Facial Expressions using Geometry Videos

Jiazhi XIA
Nanyang Technological University

Ying HE
Nanyang Technological University

Dao T. P. QUYNH
Nanyang Technological University

Xiaoming CHEN
Nanyang Technological University

Steven C. H. HOI
Singapore Management University, CHHOI@smu.edu.sg

DOI: https://doi.org/10.1145/1873951.1874010

Modeling 3D Facial Expressions Using Geometry Videos

Jiazhi Xia Ying He Dao T. P. Quynh Xiaoming Chen Steven C. H. Hoi
School of Computer Engineering
Nanyang Technological University
50 Nanyang Avenue, BLK N4, Singapore, 639798
{xiaj0002|yhe|daot0006|xmchen|chhoi}@ntu.edu.sg

ABSTRACT

The significant advances in developing high-speed shape acquisition devices make it possible to capture the moving and deforming objects at video speeds. However, due to its complicated nature, it is technically challenging to effectively model and store the captured motion data. In this paper, we present a set of algorithms to construct geometry videos for 3D facial expressions, including hole filling, geodesic-based face segmentation, and expression-invariant parametrization. Our algorithms are efficient and robust, and can guarantee the exact correspondence of the salient features (eyes, mouth and nose). Geometry video naturally bridges the 3D motion data and 2D video, and provides a way to borrow the well-studied video processing techniques to motion data processing. With our proposed intra-frame prediction scheme based on H.264/AVC, we are able to compress the geometry videos into a very compact size while maintaining the video quality. Our experimental results on real-world datasets demonstrate that geometry video is effective for modeling the high-resolution 3D expression data.

Categories and Subject Descriptors

I.3.5 [**Computational Geometry and Object Modeling**]:

General Terms

Algorithms, Design

Keywords

Geometry video, motion data, 3D facial expression, video compression, H.264/AVC, motion data parametrization, feature correspondence

1. INTRODUCTION

Over the past decade, we have witnessed a revolution in movie and game industries resulting from the use of motion data. Nowadays, it is very common that actors work in front of a blue screen and interact with invisible computer animated characters which are added later, trying to fit into a computer animated world. The movements of actors are recorded using a motion capture (or mocap) system, by which complex movement, realistic physical interactions, and exchange of forces can be recreated in a physically accurate manner. Despite the great success in movies and gaming, the current motion capture usually requires the subject to wear calibrated markers. The output of motion capture is just the approximate motion of a skeleton representing the rigid parts of the subject, rather than its precise geometry. Therefore, much editing work is often needed to map the skeletal movement to a virtual character. Furthermore, artifacts may occur when applying the recorded motion to a virtual model with proportions different than the captured subject.

The latest 3D image sensing technology provides an alternative way to capture the moving and deforming objects. For example, the structure light technique is based on wave optics, by encoding phase information of the light by light intensity. All objects in the scene are then arranged in layers according to the distance information sensed by the depth pixels in the camera, providing depth information in real time. The system consists of a structured light source (such as a digital projector) and a high speed digital video camera, and can be set up easily in an everyday environment.

Compared to the traditional marker based mocap system, the 3D camera provides us a way to capture the moving objects in a less restrictive manner, i.e. without placing any markers on the subject, and it can provide more accurate geometry data of the objects. However, the current structure light based 3D camera has several serious drawbacks that inhibits its use in broader applications:

- First, the scanned motion data is usually bulky. For example, the cutting-edge high-resolution 3D camera [38] is able to capture 30 fps with a resolution of 512×512 of each frame, approximately 4.88MB raw data per frame, 878MB per second and 51.44GB per minute as shown in Fig. 1. This imposes a challenge for compressing the captured video with a high compression ratio while maintaining the video reconstruction quality.

- Second, the captured raw data may contain noise and/or holes due to various reasons, such as camera occlusion, specular reflection, shadows, light interference, depth discontinuity, etc. Thus, much efforts are needed to clean and repair the datasets.

MM'10, October 25–29, 2010, Firenze, Italy.

- Third, each frame of the captured motion data is in the reference system of the scanner, and it is not registered in object space. Thus, the correspondences between points in different frames are not available.

(a) (b) (c)

Figure 1: The 3D camera is capable of capturing high-resolution motion data at 30 fps. Both geometry (vertex coordinate) and texture (greyscale color) are encoded in a quadrilateral mesh with approximately 250K vertices. (a) The image captured by conventional 2D camera. (b) The 3D mesh captured by 3D camera. (c) Another view of the 3D mesh.

Geometry video (GV) is a novel concept that bridges 3D motion data and video, and provides a way to borrow the well-studied video processing techniques to motion data compression and processing. However, the existing GV techniques (e.g. [1]) applied only to datasets that are created by the animators, of which the correspondences among frames are available and the data are usually simple and clean.

To solve the aforementioned challenges and promote GV to real-world applications, this paper presents a novel framework that can capture high-resolution motion data in a less restrictive manner, store the recorded data in a compact way, and allow users to manage, manipulate, and render the data easily. In this paper, we demonstrate GV on 3D human expressions. Given the captured expression data, GV first analyzes the geometry and detects salient features, and then parameterizes the motion data to a rectangular domain such that the detected features in all frames can be mapped consistently. Finally, the parameterized motion data are converted into a video format such that the well-developed video compression techniques can be used to compress the motion data. Specifically, the GV compression task in this work will be accomplished by the state-of-the-art video coding standard - H.264/AVC [35] with our tailored intra-frame prediction scheme.

The specific contributions of this paper include:

- We present a set of algorithms to construct GV for 3D facial expressions, including hole filling, geodesic-based face segmentation and expression-invariant parametrization. Our algorithms are efficient and robust, and can guarantee the exact correspondence of the salient features (eyes, mouth and nose).
- We apply the GV framework to both real-world facial expression data and synthetic motion data. By taking advantage of the strong spatial coherence of GV, we present a tailored intra-frame prediction scheme for GV in addition to that in original H.264/AVC. Our experimental results show that the proposed framework is highly effective to model 3D motion data into a very compact size while maintaining high video quality.

The rest of this paper is organized as follows. Section 2 briefly reviews the related previous work. Section 3 presents the overview of the proposed GV framework. Section 4 presents the 3D motion data acquisition and pre-processing. Section 5 details the algorithm to parameterize the motion data. Section 6 presents our proposed "tailored H.264/AVC" for GV compression. Experimental results are presented in Section 7. Finally, we conclude our work and highlight the future work in Section 8.

2. RELATED WORK

GV bridges two different research fields, geometry processing and video processing. This section briefly reviews the related work in motion data acquisition and processing, geometry images/videos, and video compression.

2.1 3D motion data acquisition and processing

In recent years, we have witnessed the significant advances in developing high-speed shape acquisition devices. Using range scanning techniques, such as phase-shifting structure light [21, 12, 38] and spacetime stero [18, 37], it is possible to scan high-resolution 3D geometry and/or texture of moving and deforming objects at video speeds.

Wang *et al.* presented a data-driven approach for accurate facial tracking and expression retargeting [34]. Wang *et al.* simplified the 3D human face registration problem to a 2D image matching problem by conformal parametrization [32]. Mitra *et al.* proposed an algorithm to register large sets of unstructured point clouds of moving and deforming objects without computing correspondences [16]. Chang and Zwicker presented an unsupervised algorithm that aligns a pair of articulated shapes with significant motion and missing data [3]. Sharf *et al.* developed a volumetric space-time technique to reconstruct the moving and deforming objects from point clouds [22]. Wang *et al.* developed an efficient non-rigid 3D motion tracking algorithm to establish inter-frame correspondences that facilitate the temporal study of subtle motions in facial expressions [33].

Observing that the human facial expressions are isometric, Bronstein *et al.* developed an algorithm to embed human faces into spherical domain, by which the canonical spherical coordinates induce an expression-invariant parametrization [2]. In this paper, we also present an expression-invariant parametrization algorithm. Our method is different than [2] in following aspects: 1) our algorithm guarantees the exact correspondence of the salient features (eyes, mouth and nose); and 2) the parametrization distortion is much less than that of [2]. To our best knowledge, this is the first work that can parameterize the 3D facial expressions with guaranteed feature correspondence.

2.2 Geometry images and videos

The concept of geometry images was pioneered by Gu *et al* [6], who parameterized the 3D mesh into a square domain and then encoded the normalized vertex coordinates (x, y, z) as a pixel value (r, g, b) of a 2D image. Therefore, geometry images naturally bridge 3D shape compression and 2D image compression algorithms, e.g. [9]. Along this direction, Lin *et al.* [13] presented JEPG2000 for compression and streaming of geometry images. Peyré and Mallat presented geometric bandlets to compress geometry images and normal maps [20]. They showed that bandeletization algorithm

outperforms the wavelet-based compression by removing the geometric redundancy of orthogonal wavelet coefficients.

Geometry images are an elegant representation of static shape. To model motion data, it is a natural idea to extend geometry images to geometry videos. In [1], Briceño *et al.* [1] parameterized the animated mesh sequence onto a rectangular domain and then formed geometry video. However, their method [1] applied only to synthetic data, of which the correspondence among frames are available. They also used 2D wavelet-based video compression techniques. In contrast to [1], our proposed parametrization algorithm works for real-world datesets which may contain artifacts such as holes and noise, and do not have the correspondence between adjacent frames. Furthermore, our parametrization method matches the salient features among frames in a consistent manner. As a result, the generated geometry videos are highly correlated in both spatial and temporal domains. This feature enables us to exploit the potential of H.264/AVC, which is incorporated with many advanced video compression techniques, for heavier compression of GV.

2.3 Video compression and H.264/AVC

Video compression aims at reducing the amount of data used to represent the video information. Traditional 2D video compression techniques can be categorized as prediction, transformation, quantization and entropy coding [25]. Prediction will produce a set of predicted values so that some video information can be represented as only the differences (residuals) from the predicted values, e.g. intra-frame prediction [35] and inter-frame motion estimation [25]. Transformation will transform pixel values or residuals into another domain so that the significant visual information is concentrated into a small number of coefficients, e.g. the Discrete Cosine Transform (DCT) [17]. Quantization [25] will reduce the representation precision of pixel values or residuals, e.g. rounding off the less significant video information. Entropy coding, e.g. the well-known Huffman coding [25], is to compress the symbols representing the video information by taking into account the possibilities of their occurrences.

In this work, we compress GV by using the H.264/AVC, and in particular we would also investigate better intra prediction scheme of H.264/AVC for GV. In H.264/AVC, an intra-frame is compressed by using intra-frame prediction [35], which allows the video encoder to predict pixel values of the current block from its previously reconstructed upper and left neighbor pixels. There is also a considerable amount of intra-frame prediction schemes proposed in recent years, e.g. [27, 19, 28, 11] focusing on reducing the prediction complexity, and [39, 40, 14, 31] aiming at reducing the prediction errors. However, all of the above schemes are mainly designed for compressing natural video pictures. In this work, we will present a tailored intra-frame prediction scheme for our GV framework in Section 6.

3. SYSTEM OVERVIEW

This section briefly shows the pipeline of the GV framework that contains the following three steps.

- Step 1 - Data acquisition and pre-processing: Using a structure-light based 3D camera system, we can capture the high resolution 3D expressions in real time. However, the captured raw data usually contains noise, holes and other artifacts. In the pre-processing step, we first fill the both the geometry and texture of holes by constructing a minimal surface with C^1 continuity along the hole boundaries. Then we track the salient facial features, such as eyes, nose, mouth, etc using active appearance modeling (AAM). Next, we segment the facial expression by computing a geodesic mask that is invariant to the expressions. Finally, we remove the eyes and mouth. See Section 4.

- Step 2 - Motion data parametrization: Each frame of a GV sequence is a 3D mesh with its own resolution and tessellation. It is highly desirable to map them to a canonical domain such that the expressions can be re-sampled to the same triangulation. Parametrization serves this purpose. Our motion data parametrization can guarantee the exact feature correspondence. See Section 5.

- Step 3 - Geometry video compression: Using the motion data parametrization technique, we map the 3D facial expressions to the rectangular domain and then construct the GV. Utilizing our tailored intra-frame prediction scheme and the powerful H.264/AVC compression tools, we can compress the GV into a very compact size while maintaining the detailed 3D motion data. See Section 6.

4. 3D MOTION DATA ACQUISITION AND PRE-PROCESSING

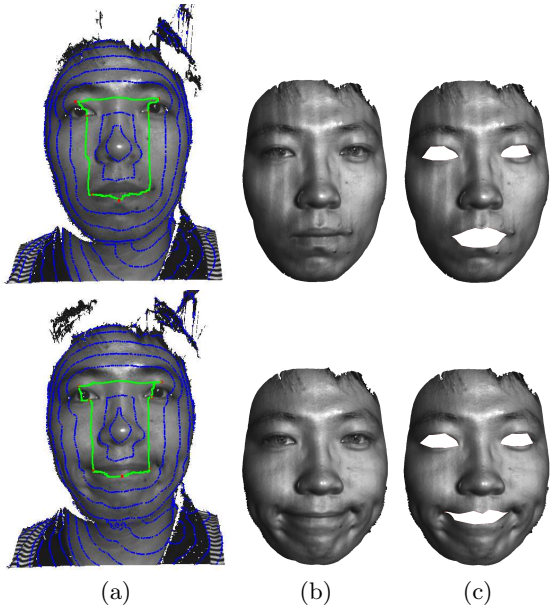


Figure 2: Face segmentation using geodesic mask. Human expressions are approximate isometry, thus, the geodesic distance is independent of the expressions. We first compute a geodesic mask from the detected features on mouth and eyes (see (a)), then segment the front face by the user-specified radius (see (b)). Finally, we remove the mouth and eyes (see (c)).

We employ the structure light-based 3D camera system [38] to capture the moving objects in real time. The system contains a video camera and a structured light projector. The projector projects digital fringe patterns composing of vertical straight stripes to the object. The stripes are deformed due to the surface profile. Then a high-speed CCD camera synchronized with the projector captures the distorted fringe image. Finally, by analyzing the fringe images, the 3D information is obtained based on the deformation using triangulation. The system is able to capture the geometry and texture of the moving objects in real time. Although very fast, the 3D camera system is not robust due to various reasons, such as ambient light interference, occlusions, shadows, and depth discontinuity. Therefore, much efforts are needed to pre-process the captured raw data.

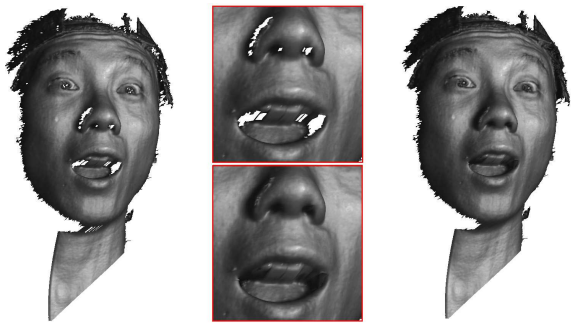


Figure 3: The captured raw data usually contains holes due to the occlusions. We fill both the geometry and texture of the holes by constructing a minimal surface which has C^1 continuity along the hole boundaries.

Feature tracking We first project the captured 3D expressions to 2D images, and then denote the salient features, including nose tip, eyes, mouth, eyebrows, etc, on the first frame. Next we use Active Appearance Model (AAM) [4] to track the feature points for the remaining frames automatically. Finally, the 2D feature points are mapped back to the 3D meshes.

Hole filling The captured raw data is a genus-0 open surface M. Let $\partial M = \gamma_0 \cup \gamma_1 \cup \cdots \gamma_k$ denote boundaries where γ_0 is the outer boundary and γ_i, $i \geq 1$ the interior holes. To fill the hole γ_i, we construct a minimal surface H_i that satisfies the following Laplacian equation [7]:

$$\triangle v = 0, \quad \forall v \notin \partial H_i \tag{1}$$

with boundary conditions

$$v|_{\partial H_i} = v|_{\gamma_i} \tag{2}$$

$$\nabla v|_{\partial H_i} = \nabla v|_{\gamma_i}. \tag{3}$$

The boundary conditions guarantee that the filled surface is of C^1 continuity along γ_i, thus, leads to visually pleasing results. Note that we can also fill the colors using the same equation except that the vertex position (x, y, z) is replaced by the color (r, g, b). Figure 3 shows the hole filling results.

Face segmentation The captured raw data contains not only the 3D faces, but also some unnecessary information, such as the cloth, hair, and background. Observe that the human expressions are approximate isometry, thus, the intrinsic properties, such as Gaussian curvature, first fundamental form, geodesic, conformal factor, etc, which are invariant under isometry, can be used to segment the face. In our framework, we adopt the geodesic since it is fairly easy to compute and highly robust to the mesh resolution and triangulation. With the eyes and mouth as source points, we compute the "multiple-sources all-destinations" geodesic using the modified Xin and Wang's algorithm [36] which takes only a few seconds for each frame. Then we segment the facial expressions using the user-specified radius. Finally, we remove the eyes and mouth. As shown in Fig. 2, our method leads to highly consistent segmentation results.

5. PARAMETERIZING MOTION DATA

In each frame of the captured motion data, the geometry is given in the reference system of the scanner, and it is not registered in object space, and correspondences between points in different frames are not available. From the analysis and editing point of view, it is highly desirable to find the correspondence among the captured data. Motion data parametrization serves this purpose by mapping all frames to a parametric domain and then re-sample the data on the domain.

Although there are large amount of literatures in surface parametrization [5] [23], there is little work on the motion data parametrization. The key challenging in motion data parametrization is that it must take the temporal coherence into consideration, i.e., the features in all frames should be mapped consistently to the parametric domain.

This section presents a novel algorithm to parameterize the 3D facial expression data. The proposed algorithm is guaranteed to be bijective and the salient facial features (such as mouth, nose and eyes) are mapped consistently on the parametric domain.

The input of our algorithm is a sequence of genus-0 meshes with four boundaries. Let M denote the mesh and $\partial M = \gamma_0 \cup \cdots \cup \gamma_3$ are the boundaries, where γ_0 is the boundary of the human face, γ_1 and γ_2 the two eyes and γ_3 the mouth. We design the parametric domain $D \in \mathbb{R}^2$ as the rectangle with three holes, thus, D has the same topology of M.

We first compute the geodesic c between two eyes, i.e., γ_1 and γ_2. Then we compute the geodesic d from the middle point of c to the mouth γ_3. Note that geodesic is an intrinsic property, thus, independent of the expressions which are approximate isometry. By slicing the mesh along the geodesics c and d, the number of boundaries is reduced to 2. The resulted mesh $\overline{M}$ is a genus-0 mesh with two boundaries, i.e., γ_0 and $\gamma_1 \cup \gamma_2 \cup \gamma_3 \cup c \cup d$. In the following, we use $\partial\overline{M}_0$ and $\partial\overline{M}_1$ to denote the two boundaries of $\overline{M}$.

Then we compute the harmonic function $f : \overline{M} \to \mathbb{R}$,

$$\triangle f(v) = 0, \quad \forall v \notin \partial\overline{M},$$

with Dirichlet boundary condition:

$$f(v) = 0, \quad \forall v \in \partial\overline{M}_0,$$

$$f(v) = 1, \quad \forall v \in \partial\overline{M}_1.$$

Since the function f is harmonic, all its local extrema are on the boundaries. Furthermore, the mesh $\overline{M}$ is of genus-0 with two boundaries. According to Morse theory [15], f has no critical point (the point with vanishing gradient)

inside $\overline{M}$. Therefore, the gradient vector field ∇f has no singularity. The integration curve of ∇f is a curve such that the tangent vector to the curve at any point v along the curve is precisely the vector $\nabla f(v)$. In the Appendix, we show that each integral curve has unique ending points, one on $\partial\overline{M}_0$, the other on $\partial\overline{M}_1$. Furthermore, any two integral curves do not intersect.

We process the parametric domain D in the same way and let $\overline{D}$ denote the sliced mesh with two boundaries. We compute the harmonic function $g : \overline{D} \to \mathbb{R}$ with the same boundary condition as f. We also construct a bijective map between two boundary curves $h : \partial\overline{M}_1 \to \partial\overline{D}_1$ by arc-length parametrization.

Then the parametrization $\phi : \overline{M} \to \overline{D}$ is constructed as follows: For each vertex $v \in \partial\overline{M}_1$, trace the integral curve $\alpha \in \overline{M}$ following the gradient ∇f. Then, starting from $h(v) \in \partial\overline{D}_1$, trace another integral curve $\beta \in \overline{D}$. Thus, we build a one-to-one map between two integral curves α and β. By going through every point $v \in \partial\overline{M}_1$, we build the one-to-one map between $\overline{M}$ and $\overline{D}$ which in turns induces a one-to-one map $\phi : M \to D$.

Input:
$M \in \mathbb{R}^3$, the input 3D facial expression of genus-0 mesh with four boundaries, $\partial M = \gamma_0 \cup \cdots \cup \gamma_3$;
$D \in \mathbb{R}^2$, the parametric domain with the same topology of M;
Output:
The one-to-one map $\phi : M \to D$ such that the salient features (eyes, mouth and nose) are mapped to the corresponding features on D.

1. *Compute the geodesic c between γ_1 and γ_2.*
2. *Compute the geodesic d from the middle point of c to γ_3.*
3. *Cut M along c and d, the resulted mesh $\overline{M}$ is of genus-0 with 2 boundaries.*
4. *Process the parametric domain D in the similar way (as steps 1 to 3). Let $\overline{D}$ denotes the processed mesh of genus-0 with 2 boundaries.*
5. *Compute the harmonic function $f : \overline{M} \to \mathbb{R}$ with Dirichlet boundary condition, $\triangle f = 0$, $f|_{\partial\overline{M}_0} = 0$, $f|_{\partial\overline{M}_1} = 1$*
6. *Compute the harmonic function $g : \overline{D} \to \mathbb{R}$ with Dirichlet boundary condition, $\triangle g = 0$, $g|_{\partial\overline{D}_0} = 0$, $g|_{\partial\overline{D}_1} = 1$*
7. *Parameterize $\partial\overline{M}_1$ and $\partial\overline{D}_1$ by the arc length parametrization, $h : \partial\overline{M}_1 \to \partial\overline{D}_1$.*
8. *For each point $v \in \partial\overline{M}_1$*
8.1 *Trace the integral curve $\alpha \in \overline{M}$ of the gradient vector field ∇f.*
8.2 *Trace another integral curve $\beta \in \overline{D}$ starting from $h(v) \in \partial\overline{D}_1$ and following the vector field ∇g.*
8.3 *Construct the one-to-one map $\overline{\phi} : \overline{M} \to \overline{D}$ as $\overline{\phi}(\alpha) = \beta$*
9. *The parametrization $\phi : M \to D$ is induced from $\overline{\phi} : \overline{M} \to \overline{D}$.*

Algorithm 1: Expression-invariant 3D face parametrization

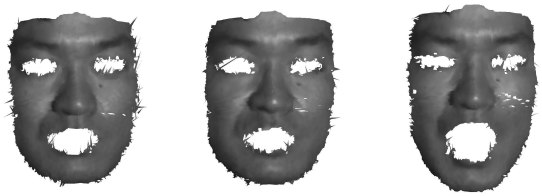
Remark In the parametrization algorithm, we cut the 3D face along the geodesics connecting the three holes (i.e., eyes and mouth). So the resulted mesh is of genus-0 with 2 boundaries. The inner boundary $\gamma_1 \cup \gamma_2 \cup \gamma_3 \cup c \cup d$ is invariant to the expressions, thus, highly consistent among all frames. Furthermore, the outer boundary is determined by a geodesic mask with the user-specified radius applied to all expressions. Thus, the outer boundary is also invariant to the expression. Observe that the harmonic function is intrinsic to the geometry and independent of the expressions. As a result, the proposed parametrization is invariant to the expression. Furthermore, as proven in the Appendix, the inner boundary of $\overline{M}$ is mapped to the inner boundary of $\overline{D}$ precisely, thus, guarantees the exact correspondence of salient features, such as eyes, mouth and nose. As shown in Fig. 4, two expressions are parameterized consistently using our approach.

Using the expression invariant parametrization, we can parameterize a sequence of facial expressions to the canonical parametric domain with guaranteed correspondence of the salient features (mouth, nose and eyes).

(a) Original data

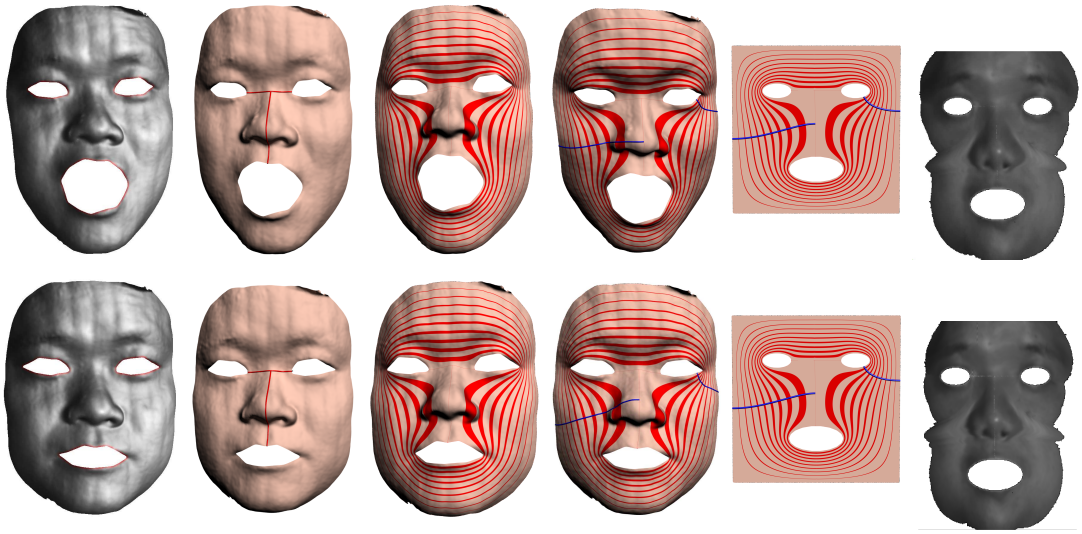
(b) Compression ratio 21:1, PSNR 71.73dB

(c) Compression ratio 212:1, PSNR 58.31dB



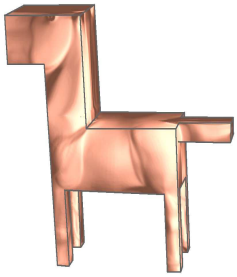
(d) Compression ratio 575:1, PSNR 46.00dB

Figure 6: Geometry video of HumanFaceGV1. (a) shows the original video sequence; (b), (c) and (d) show the reconstructed videos at low, medium and high compression ratios by using the tailored H.264/AVC.

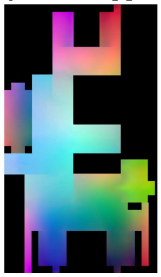


(a) (b) (c) (d) (e) (f)

Figure 4: Expression-invariant parametrization. (a) Input mesh M. (b) Geodesics connecting the eyes, nose and mouth. (c) Harmonic function with Dirichlet boundary condition. (d) Integral curves follow the gradient of the harmonic function. (e) Integral curves on the parametric domain. (f) The integral curves induce the parametrization between M and D, which guarantees the exact correspondence of the eyes, mouth and nose.



(a) Polycube mapped horse



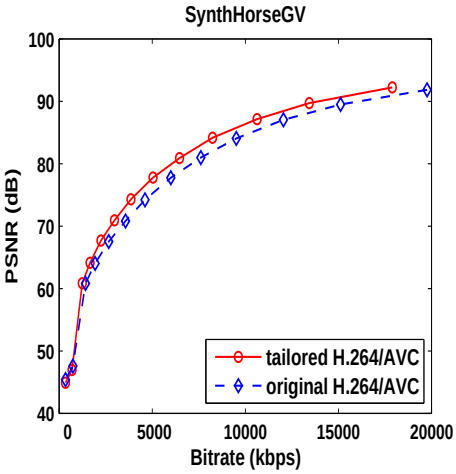
(b) Rectangular parametric domain

(c) Original data

(d) Compression ratio 48:1
PSNR 81.05dB

(e) Compression ratio 140:1
PSNR 67.68dB

(f) Compression ratio 466:1
PSNR 47.98dB



(g) Rate-distortion performance
of tailored H.264/AVC
in comparison to original H.264/AVC

Figure 5: Geometry video of SynthHorseGV. (a) maps the surface of horse onto polycube. (b) parameterizes the mapped polycube into rectangular domain. (c) shows the original video sequence. (d), (e) and (f) show the reconstructed videos at low, medium and high compression ratios by using our tailored H.264/AVC. (g) shows the rate-distortion (PSNR vs. Bitrates) performance of our tailored H.264/AVC in comparison to the original H.264/AVC.

6. GEOMETRY VIDEO COMPRESSION

Using the proposed motion data parametrization algorithm, each frame of the motion data is parameterized to a rectangular domain that can be easily converted into a geometry image, i.e. the pixel color R, G, and B representing the vertex coordinates, x, y, and z. Then the 2D GV is constructed by combining all geometry images together.

The temporal feature of GV is quite similar to that of natural videos, i.e. a block in the current video picture usually closely matches another block locating at the same or close position in the neighbor picture. As a result, the temporal redundancy in GV can be significantly removed by using existing motion estimation algorithms [35].

The spatial feature of GV, on the other hand, is different from that of natural video pictures. Since the pixel values of a GV picture are actually the vertex coordinates of the 3D model in the 3D space, the adjacent pixels usually share a correlated local variant pattern. For example, assuming that the 3D model surface is relatively smooth in a small local region, e.g. a 4x4 block region, the corresponding pixels within that region will be smoothly-varying too. In this work, we model and utilize this feature of GV by proposing a simple yet effective "tailored intra-prediction scheme" for better compression.

Specifically, we have added 4 extra intra-frame prediction modes to the original H.264/AVC in our tailored scheme. These extra modes will form predicted pixels by considering the variant pattern of neighbor pixels. For a 4x4 block in a video picture, our proposed prediction modes can be written as:

$$P(x,y)=\begin{cases} P(x,y-1)+[N(x-1,y-1)-N(x-1,y)], & \text{mode 1} \\ P(x,y-1)+\frac{1}{4}\sum_{i=1}^{4}[N(x-1,y+i)-N(x-1,y+i-1)], & \text{mode 2} \\ P(x-1,y)+[N(x,y-1)-N(x-1,y-1)], & \text{mode 3} \\ P(x-1,y)+\frac{1}{4}\sum_{i=1}^{4}[N(x-1,y+i)-N(x+i-1,y-1)], & \text{mode 4} \end{cases}$$

where $N(x,y)$ denotes the neighbor pixel locating at position (x,y) and $P(x,y)$ is the predicted pixel in current block.

The tailored H.264/AVC encoder will then select the best prediction mode from the original H.264/AVC modes and our extra modes based on minimum prediction errors.

These extra prediction modes were only used for 4x4 blocks in a video picture. The extra prediction modes may slightly impose the computational overhead on the encoder. However, the overhead is neglectable compared with those higher-complexity components in H.264/AVC, e.g. motion estimation.

In the next section, we will show that the tailored prediction scheme yields better prediction results, leading to obtain lower bitrates of the encoded video.

7. EXPERIMENTAL RESULTS

In this section, we present the results of compressing and reconstructing the 3D motion data by using our GV framework.

7.1 Experimental setups

7.1.1 Test GV sequences

There are 4 test GV sequences of human faces (named "HumanFaceGV1" to "HumanFaceGV4") captured by our camera from 4 subjects, who were asked to make varied facial expressions, e.g. laughing, shouting, etc, during the capture. Each of the sequence consists of 200 frames with the resolution of 320x320. For each video sequence, we compress it at different bitrates and then decompress and reconstruct the video. The peak signal to noise ratio (PSNR) is used to measure the reconstruction quality of the video.

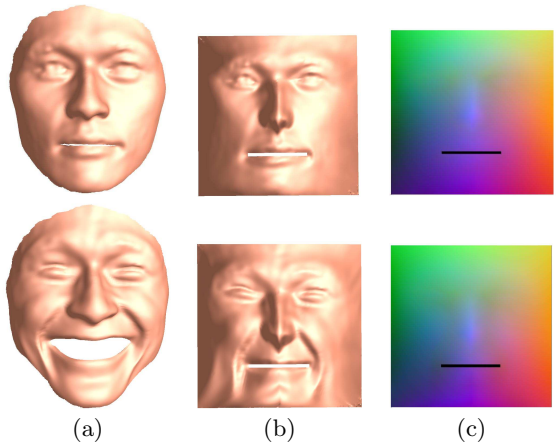


Figure 8: Geometry video of a synthetic face model (SynthFaceGV). (a) 3D model. (b) Parameterization. (c) Geometry image (video frame of a GV).

In addition to the real human expressions, we also tested 2 sequences containing synthetic 3D motion data, including a synthetic face model and a horse gallop model, which were named as "SynthFaceGV" and "SynthHorseGV" respectively.

The motion in SynthHorseGV is composed of a few rigidly moving parts, such as the horse gallop shown in Fig. 5. In contrast to our camera-captured facial expression datasets which can be parameterized to the rectangular domain directly, the horse model is a closed surface with complex geometry. To reduce the parametrization distortion, we first parameterized it to a polycube domain which mimics the geometry of the model (see Fig. 5(a)). Then we flattened the polycube parametrization to the 2D rectangular domain (see Fig. 5(b)). As the correspondence is available for these synthetic animation datasets, we only need to parameterize the first frame. In our implementation, we adopted the algorithm presented in [8] to parameterize the horse model.

The SynthFaceGV is topologically equivalent to a annulus, we applied our constrained parametrization such that outer and inner boundaries are mapped to the unit square and a slim rectangle respectively (see Fig. 8). We should also point out that the synthetic GVs do not contain the texture information.

7.1.2 H.264/AVC configurations

In this work, the proposed "tailored intra-frame prediction scheme" (as described in Section 6), was implemented based on the JM14.2 H.264/AVC reference software [10]. The High 4:4:4 profile of H.264/AVC FREXt [26] was adopted in this work to keep the high fidelity of the motion data, i.e. each color channel of a GV picture is equally compressed and there is no subsampling used. The test GV sequences were encoded at 30 frames per second, with Group of Picture (GOP) structure of "IPBPB", Fast Full Search motion estimation (32x32 search range, 5 reference frames and variable block sizes) and CABAC entropy coding. Rate-Distortion

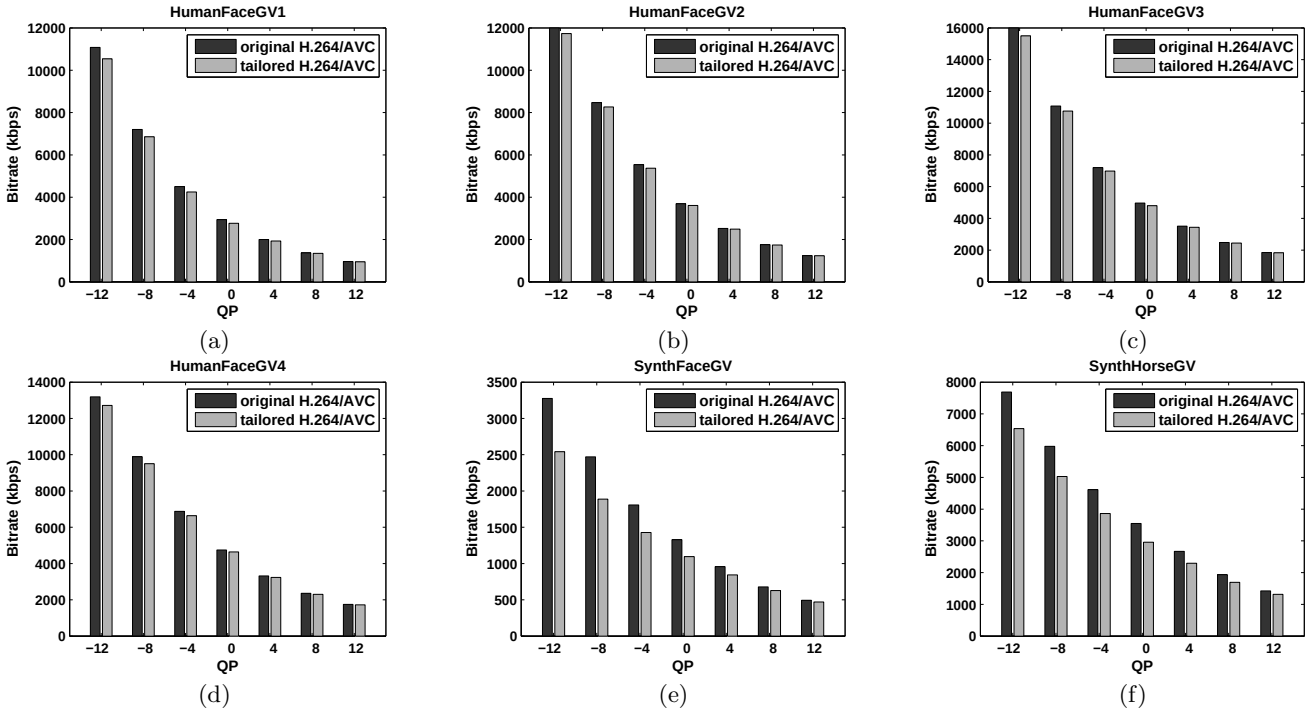


Figure 7: Bitrates of our tailored H.264/AVC in comparison to the original H.264/AVC for QP ranging from -12 to +12, where a lower QP results in a higher video quality but a lower compression ratio: (a) HumanFaceGV1. (b) HumanFaceGV2. (c) HumanFaceGV3. (d) HumanFaceGV4. (e) SynthFaceGV. (f) SynthHorseGV.

Optimization (RDO) was turned on. The intra-frame period (interval) was set to 30 (i.e. 1 intra-frame per second). The GVs were compressed at different bitrates by adjusting the quantization parameters (QPs). Please refer to [10] for more details about the above encoding parameters and configurations.

7.2 Results

Fig. 7 illustrates the bitrates achieved by the modified H.264/AVC with our tailored intra-frame prediction scheme (abbreviated as "tailored H.264/AVC" in the remaining parts), in comparison to those achieved by original H.264/AVC. The QPs range from -12 to +12, where a lower QP results in a higher video quality but a lower compression ratio. Please note that we have adopted some negative QPs in order to encode high-quality GVs [24]. It has been observed that our tailored H.264/AVC outperforms the original H.264/AVC, e.g. for HumanFaceGVs, the tailored H.264/AVC is able to further reduce the bitrate of original H.264/AVC by 1.02% to 5.56% while increasing the PSNR by 0.04dB to 0.22dB at the same time. For the synthetic GVs (SynthFaceGV and SynthHorseGV), the bitrate reduction obtained by our tailored H.264/AVC over original H.264/AVC is up to 20%, which is more significant than those for HumanFaceGVs. This is because that these synthetic GVs are articulated animation containing less noise. Also, the surfaces of the synthetic models are smoother compared to HumanFaceGVs thus enabling better prediction results of our tailored H.264/AVC. The above results demonstrated that our tailored H.264/AVC can better utilize the spatial feature of GV.

Fig. 6 (a)-(d) display sample frames of the original HumanFaceGV1 and the reconstructed video by our tailored H.264/AVC with different compression ratios. We have observed that the tailored H.264/AVC will result in no visual distortion at low compression ratio (21:1). At medium compression ratio (212:1), the decoded frames showed a acceptable quality but small deformation can be found in the boundary of the human face. At very high compression ratio (575:1), the decoded frames are visually distorted. Fig. 5 (a)-(f) shows sample frames of original SynthHorseGV and the reconstructed video sequence. We observed from this figure the similar results to HumanFaceGVs, i.e. the reconstructed video frames are of good quality at low and medium compression ratios. Fig. 5 (g) demonstrates the rate-distortion performance of our tailored H.264/AVC is significantly better than the original H.264/AVC for SynthHorseGV.

As a summary, our experimental results show that the GVs can be significantly compressed by using H.264/AVC. By employing our tailored H.264/AVC, a further improvement over original H.264/AVC can be achieved in both bitrate reductions and PSNR gains, resulting in a good compression ratio (over 200:1) without introducing visual artifacts.

8. CONCLUSIONS AND FUTURE WORK

This paper presented a novel framework to model 3D facial expressions using geometry video (GV). Within our framework, we parameterize the 3D motion data with guaranteed feature correspondence and store them into a video format. Our investigation shows that the GV framework is very effective for modeling the 3D motion data. In particu-

lar, it allows the 3D motion data being heavily compressed by using well-studied video compression techniques. Our experimental results on real-world datasets show that the GV can be significantly compressed by H.264/AVC. By taking advantage of the strong spatial coherence of GV, we have also presented a tailored intra-frame prediction scheme and incorporated it into H.264/AVC. The experimental results demonstrated that our tailored scheme can achieve bitrate reductions and PSNR improvement concurrently over the original H.264/AVC. In conclusion, our proposed GV framework enables efficient modeling, storage, and manipulation of the high-resolution 3D motion data thus providing an attractive way in multimedia information processing in this regard.

There are several future research directions. First, this paper focuses on the 3D facial expression due to its simplicity. The general 3D motions of human and animals are usually of complex geometry and topology. Thus, to reduce the parametrization distortion, a polycube is an ideal parametric domain as we demonstrated in the horse gallop model. However, the existing polycube parametrization algorithms such as [30][29] do not allow the users to freely specify the key feature correspondence among frames. Second, there is a need to develop an automatic method to flatten the polycube into the planar domain in a space efficient manner. Third, our current tailored intra-frame prediction scheme has not fully exploited the potentials of dedicated compression techniques for GV. These issues will be further investigated in our future work.

9. ACKNOWLEDGMENTS

This work was fully supported by Singapore NRF Interactive Digital Media R&D Program, under research grant NRF2008IDM-IDM004-006. The authors would like to thank the anonymous reviewers and the paper shepherd, Prof. Wei Tsang Ooi, for their constructive comments and suggestions in improving the final version of the paper.

10. REFERENCES

[1] H. Briceño, P. Sander, L. McMillan, S. Gortler, and H. Hoppe. Geometry videos: a new representation for 3d animations. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 136–146. Eurographics Association, 2003.

[2] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Expression-invariant representations of faces. *IEEE Transactions on Image Processing*, pages 537–547, 2007.

[3] W. Chang and M. Zwicker. Automatic registration for articulated shapes. *Computer Graphics Forum*, 27(5):1459–1468, 2008.

[4] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *TPAMI*, 23(6):681–685, 2001.

[5] M. S. Floater and K. Hormann. Surface parameterization: a tutorial and survey. In *Advances in multiresolution for geometric modelling*, pages 157–186. 2005.

[6] X. Gu, S. J. Gortler, and H. Hoppe. Geometry images. *ACM Trans. Graph.*, 21(3):355–361, 2002.

[7] X. Gu and S.-T. Yau. *Computational Conformal Geometry*. International Press of Boston, 2008.

[8] Y. He, H. Wang, C.-W. Fu, and H. Qin. A divide-and-conquer approach for automatic polycube map construction. *Computers and Graphics*, 33(3):369–380, 2009.

[9] H. Hoppe and E. Praun. Shape compression using spherical geometry images. In *Advances in Multiresolution for Geometric Modelling*, pages 27–46. Springer-Verlag, 2005.

[10] ITU-T. The h.264/avc standard. *ITU-T Rec. H.264*, 2008.

[11] D.-Y. Kim, K.-H. Han, and Y.-L. Lee. Adaptive single-multiple prediction for h.264/avc intra coding. *TCSVT*, 2010.

[12] T. P. Koninckx and L. Van Gool. Real-time range acquisition by adaptive structured light. *TPAMI*, 28(3):432–445, 2006.

[13] N.-H. Lin, T.-H. Huang, and B.-Y. Chen. 3d model streaming based on a jpeg 2000 image. *Consumer Electronics, IEEE Transactions on,*, 53(1):182–190, 2007.

[14] D. Liu, X. Sun, F. Wu, and Y.-Q. Zhang. Edge-oriented uniform intra prediction. *TIP*, 17(10):1827 –1836, oct. 2008.

[15] J. Milnor. *Morse Theory*. Princeton University Press, 1963.

[16] N. J. Mitra, S. Flöry, M. Ovsjanikov, N. Gelfand, L. Guibas, and H. Pottmann. Dynamic geometry registration. In *SGP '07*, pages 173–182, 2007.

[17] T. N. N. Ahmed and K. R. Rao. On image processing and a discrete cosine transform. *Comput., IEEE Transactions on*, C-23(1):90 –93, jan. 1974.

[18] D. Nehab, R. Ramamoorthi, J. Member-Davis, and S. Member-Rusinkiewicz. Spacetime stereo: A unifying framework for depth from triangulation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(2):296–302, 2005.

[19] M. Parlak, Y. Adibelli, and I. Hamzaoglu. A novel computational complexity and power reduction technique for h.264 intra prediction. *IEEE Transactions on Consumer Electronics*, 54(4):2006 –2014, 2008.

[20] G. Peyré and S. Mallat. Surface compression with geometric bandelets. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pages 601–608, New York, NY, USA, 2005. ACM.

[21] S. Rusinkiewicz, O. Hall-Holt, and M. Levoy. Real-time 3d model acquisition. *ACM Trans. Graph.*, 21(3):438–446, 2002.

[22] A. Sharf, D. A. Alcantara, T. Lewiner, C. Greif, A. Sheffer, N. Amenta, and D. Cohen-Or. Space-time surface reconstruction using incompressible flow. In *SIGGRAPH Asia '08*, pages 1–10, 2008.

[23] A. Sheffer, E. Praun, and K. Rose. Mesh parameterization methods and their applications. *Found. Trends. Comput. Graph. Vis.*, 2(2):105–171, 2006.

[24] G. Sullivan. On rounding, qp value origin, dynamic range, and |f|. *JVT-C136*, may 2002.

[25] G. Sullivan and T. Wiegand. Video compression - from concepts to the H.264/AVC standard. *Proceedings of the IEEE*, 93(1):18 –31, jan. 2005.

[26] G. J. Sullivan, P. Topiwala, and A. Luthra. The

H.264/AVC advanced video coding standard: Overview and introduction to the fidelity range extensions. In *SPIE conference on Applications of Digital Image Processing XXVII*, pages 454–474, 2004.

[27] A.-C. Tsai, A. Paul, J.-C. Wang, and J.-F. Wang. Intensity gradient technique for efficient intra-prediction in h.264/avc. *TCSVT*, 18(5):694 –698, 2008.

[28] A.-C. Tsai, J.-F. Wang, J.-F. Yang, and W.-G. Lin. Effective subblock-based and pixel-based fast direction detections for h.264 intra prediction. *TCSVT*, 18(7):975 –982, 2008.

[29] H. Wang, Y. He, X. Li, X. Gu, and H. Qin. Polycube splines. *Computer-Aided Design*, 40(6):721–733, 2008.

[30] H. Wang, M. Jin, Y. He, X. Gu, and H. Qin. User-controllable polycube map for manifold spline construction. In *Symposium on Solid and Physical Modeling*, pages 397–404, 2008.

[31] L. Wang, L.-M. Po, Y. Uddin, K.-M. Wong, and S. Li. A novel weighted cross prediction for h.264 intra coding. In *ICME '09*, pages 165 –168, 2009.

[32] S. Wang, Y. Wang, M. Jin, X. Gu, and D. Samaras. 3d surface matching and recognition using conformal geometry. In *CVPR (2)*, pages 2453–2460, 2006.

[33] Y. Wang, M. Gupta, S. Zhang, S. Wang, X. Gu, D. Samaras, and P. Huang. High resolution tracking of non-rigid motion of densely sampled 3d data using harmonic maps. *International Journal of Computer Vision*, 76(3):283–300, 2008.

[34] Y. Wang, X. Huang, C. su Lee, S. Zhang, Z. Li, D. Samaras, D. Metaxas, A. Elgammal, and P. Huang. High resolution acquisition, learning and transfer of dynamic 3-d facial expressions. In *Computer Graphics Forum*, pages 677–686, 2004.

[35] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the H.264/AVC video coding standard. *Circuits and Systems for Video Technology, IEEE Transactions on*, 13(7):560 –576, july 2003.

[36] S.-Q. Xin and G.-J. Wang. Improving chen and han's algorithm on the discrete geodesic problem. *ACM Trans. Graph.*, 28(4), 2009.

[37] L. Zhang, N. Snavely, B. Curless, and S. M. Seitz. Spacetime faces: high resolution capture for modeling and animation. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 548–558, 2004.

[38] S. Zhang and P. Huang. High-resolution, real-time 3d shape acquisition. In *CVPRW '04*, volume 3, page 28, 2004.

[39] Y. Zheng, P. Yin, O. Escoda, X. Li, and C. Gomila. Intra prediction using template matching with adaptive illumination compensation. In *ICIP '08*, pages 125 –128, 2008.

[40] J. Zhou, H. Zhou, and X. Yang. An interpolation method by predicting the direction of pixel texture changing trend for h.264/avc intra prediction. In *IITA '08*, volume 1, pages 884 –888, 2008.

Appendix

In this appendix, we prove that the proposed parametrization algorithm leads to a one-to-one map and guarantees the exact boundary correspondence.

Theorem Given two surfaces A and B, both are of genus-0 with 2 boundaries. Let A_i, B_i, $i = 1, 2$ denote the boundaries, i.e., $\partial A = A_0 \cup A_1$ and $\partial B = B_0 \cup B_1$. Define harmonic function $f : A \to \mathbb{R}$ on A, $\triangle f = 0$, with Dirichlet boundary condition, $f|_{A_0} = 0$ and $f|_{A_1} = 1$.

Let $C_f : A_0 \times [0, 1] \to A$ be the integral curve of the gradient field ∇f such that given an arbitrary point $v_0 \in A_0$, $C_f(v_0, 0) = v_0$, $C_f(v_0, 1) = v_1$ and $C_f(v_0, t) = v_t$, where $v_1 \in A_1$ is the other ending point and $v_t \in M$ is the point satisfying $f(v_t) = t$. Similarly, we define the harmonic function on B, $g : B \to \mathbb{R}$ and the integral curve $C_g : B_0 \times [0, 1] \to B$.

Define a homeomorphic map $h : A_0 \to B_0$ and construct the parametrization $\phi : A \to B$ by mapping the integral curve $C_f(v_0, \cdot)$ to $C_g(h(v_0), \cdot)$ for $\forall v_0 \in A_0$.

Then the map ϕ has the following properties:

- ϕ is one-to-one;
- ϕ maps the inner boundary A_0 to B_0, i.e., $\phi(A_0) = B_0$.

Proof First, we show no integral curve of ∇f or ∇g form a loop, since f or g is smooth function and its gradient vector field is curl-free.

Second, we show that no integral curve that starts and ends on the same boundary curve. The function f or g is harmonic function that does not have critical points (where the gradient vanishes) inside the surface. Thus, the function value is strictly monotonic along the integral curve. Note that all points on the same boundary curve have the same function value, so the ending points of each integral curve must be on different boundary curve.

Third, we show that two integral curves do not intersect. Without loss of generality, assume two integral curves $\gamma_1 \in A$ and $\gamma_2 \in A$ intersect at a point p. Then p is a critical point and the gradient ∇f vanishes at p. We consider two cases:

Case 1: $p \notin \partial A$ is an interior point. Since f is harmonic, the maximum and minimum must be on the boundaries. Therefore the Hessian matrix at p has negative eigenvalue values. Suppose $f(p) = s$, then according to Morse theory, the homotopy types of the level sets $f^{-1}(s-\epsilon)$ and $f^{-1}(s+\epsilon)$ will be different. At all the interior critical points, the Hessian matrices have negative eigenvalues, the homotopy type of the level sets will be changed. The changes of the homotopy type can not be canceled out. Therefore, the homotopy type of A_0 is different from that of A_1. This contradicts the given condition, since A_0 is homotopic to A_1.

Case 2: $p \in \partial A$ is on the boundary. Without loss of generality, say $p \in A_0$. Then we can glue two copies of the same surface, along A_0. And reverse the gradient field of one surface. The union of the two gradient fields give us a harmonic function field. Then there is no interior critical point on the doubled surface. p becomes one interior critical point, that leads to a contradiction.

Therefore γ_1 and γ_2 have no intersection points anywhere.

Last, we show ϕ is one-to-one. From the above, we know that for an arbitrary interior point, there is a unique integral curve passing through and intersecting on the inner and outer boundaries. The two ending points are also unique. Furthermore, the given boundary map $h : A_0 \to B_0$ is homeomorphic, thus, it induces a homeomorphism between integral curves in A and B, $C_f(v_0, \cdot) \to C_g(h(v_0), \cdot)$. The boundary A_0 is mapped to B_0 because the map ϕ restricted on A_0 is the boundary map $h : A_0 \to B_0$, i.e., $\phi|_{A_0} = h$.