Linearly Augmented Real-Time 4D Expressional Face Capture

Shu Zhang[a,b], Hui Yu[b,*], Ting Wang[c], Junyu Dong[a] and Tuan D. Pham[d]

[a]Ocean University of China, Qingdao, China. {zhangshu, dongjunyu}@ouc.edu.cn
[b]University of Portsmouth, Portsmouth, UK. hui.yu@port.ac.uk
[c]Shandong University of Science and Technology, Qingdao, China. wangting@sdust.edu.cn
[d]The Center for Artificial Intelligence, Prince Mohammad Bin Fahd University, Al Khobar, Saudi Arabia

ARTICLE INFO

Keywords:
Linear
personalised
3D expressional face
3D information extraction
CPU computation
real-time

ABSTRACT

Personalised 3D face creation has always been a hot topic in the computer vision community. Many methods have been proposed including the statistic model, the non-rigid registration and high-end depth acquisition equipment. However, in practical applications, those existing methods still have their own limitations. For example, the performance of the statistic model-based methods highly depends on the generality of the pre-trained statistic model; the non-rigid registration based methods are sensitive to the quality of input data; the high-end equipment-based methods are less able to be popularised due to the expensive equipment costs; the deep learning-based methods can only perform well if proper training data provided for the target domain, and require GPU for better performance. To this end, this paper presents an adaptive template augmented method that can automatically obtain a personalised 4D facial modelling only using a consumer-grade device. The noisy data from such a cheap device are well handled. The whole process consists of a series of linear solutions and can be achieved in real-time for online processing only based on the CPU computation on a laptop. There is no constraint nor complex operation required by the proposed method. No additional time-consumptive pre- or post-processing for the personalisation is needed. Comparisons against several existing methods demonstrate the superiority of the proposed method.

1. Introduction

Although several approaches have been proposed for 3D facial reconstruction in recent years [1, 2, 3], it is still a real challenging task to create a detailed and personalised 3D face for the individual in an unconstrained manner with a minimised computational cost. The additional dimension in the 3D face can provide richer facial information than its 2D counterpart in various applications, such as facial recognition [4], facial expression analysis [5, 6, 7, 8], and facial emotion recognition [9] among others. Currently, most of those real-time 3D facial information extraction methods either utilise a pre-defined model such as 3DMM to estimate a limited number of variations of 3D facial shapes, or require complex operations with time-consumptive pre-process for high accuracy, or rely on high-end input equipment such as massive camera array or expensive 3D depth sensor for 3D facial information sensing. Some of them even require a high-end computation device such as GPU enhanced ones to support deep learning-based methods [2, 10]. However, in practice, a complex operation may pose obstructions to users that have no experience or no condition for such operation. Time-consumptive procedures might make users boring during the process. Incorrect operations may even lead to a 3D sensing failure. Moreover, the methods based on the high-end equipment are less likely to be popularised due to its increased price. Thus, a fully automatic and real-time 3D face capturing based on a consumer-grade device is highly demanded.

To this end, this paper presents a 3D facial information extraction solution that can be achieved in real-time. The proposed method automatically captures a personalised 3D face unconstrainedly. By integrating multiple frames, a 4D facial modelling scheme can be expected with a temporal-related fourth dimension. With the proposed method, the user only needs to present his/her face in front of a consumer-grade RGB-D camera, a detailed and personalised 3D face can be modelled with only one frame from this low-priced sensor. There is no restrictions on the expressions performed by the user. If a user continuously presents his/her face before the camera, a 4D facial expressional model

*This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) (EP/N025849/1); National Natural Science Foundation of China (NSFC) (41906177, 41927805); China Postdoctoral Science Fundation (2019M652476); the Fundamental Research Funds for the Central Universities, China (201964022); the Natural Science Foundation of Shandong Province of China (ZR2018ZB0852).

*Corresponding author: Hui Yu.
ORCID(s): 0000-0002-5873-634X (S. Zhang)

can be easily generated. Furthermore, the whole capturing process is linearly achieved in real-time. According to the experiments on a consumer-grade laptop, the 4D facial capturing can achieve less than 50 ms per frame only based on CPU computations for low-quality inputs.

To achieve the proposed method, an adaptive-template augmented mesh deformation scheme is presented. As shown in Fig. 1, it consists of the procedures including: 1) acquisition of the initial target face with 3D facial landmarks from an RGB-D camera; 2) the adaptive template generation per face based on 3D facial landmarks extracted; and 3) a linear solution enhanced mesh deformation for the final 3D face model. The contributions of the proposed method are as following:

1. A fully automatic solution is proposed for 4D facial capturing. Neither manual intervention nor pre-processing is required. A new user only needs to show his/her face in front of a consumer-grade RGB-D camera with no additional constraint.
2. The low-quality and noise in the input data are well handled without loss of the high accuracy in the results. As demonstrated later in the paper, large holes commonly exist in the input data by a low-priced input device. Our method can still extract detailed and personalised 3D facial information from such low-quality inputs.
3. An adaptive template scheme is adopted, with which a better initialisation for the facial mesh non-rigid deformation is guaranteed for arbitrary expressional faces.
4. The system cost and computational cost are both low. The only input device is a commodity RGB-D camera, which can be less than a hundred GBPs. 3D facial modelling is achieved by a series of linear solutions. An expressional 3D face can be obtained by only one frame data from the RGB-D camera. Differing from other existing methods that rely on GPU for fast computation, the proposed method can easily run in real-time only based on CPU computation.

The rest of the paper is organised as follows. Section 2 discusses the related work in the field of 3D facial modelling. The overview of the proposed method is introduced in Section 3. Section 4, 5 and 6 elaborate the three main components of the proposed method in details. Section 7 demonstrates the experiment of the proposed method with performance comparisons. The paper is finally concluded in Section 8.

2. Related work

In recent years, 3D facial information sensing draws a lot of attentions in various fields. Many related approaches have been proposed. They are achieved from following four aspects: (**a**) the statistic model-based solutions; (**b**) the solutions based on the non-rigid registration from a template to a target face; (**c**) the solutions using a special or high-end data acquisition equipment; and (**d**) the solutions enhanced by recent deep learning networks.

Statistic model-based solutions. The methods based on the 3D Morphable Model (3DMM) or the Blendshapes are the typical approaches that utilise the statistic models for 3D facial reconstruction. The 3DMM aims at reconstructing the 3D shape of a human face, while most Blendshapes based methods focus on the 3D facial expression reconstruction. The principle of those statistic model-based methods is a linear approximating from a series of pre-acquired 3D face bases to the target face. The 3D face basis components for the 3DMM are obtained by a Principal Component Analysis (PCA) from a large sample of 3D face scans of different people, commonly with a neutral expression. Commonly, blendshape based methods employ the 3D face bases of different expressions. Many researches have presented their efforts in those statistical model-based methods. For example, Hang *et al.* [11] presented a 3DMM for craniofacial shape and texture variation. They built the 3DMM from more than 1200 distinct identities in the Headspace dataset. Claudio *et al.* [12] presented a dictionary learning-based 3DMM for modelling. Their DL-3DMM was constructed by learning a dictionary of the basis components on the aligned facial scans. Bernhard *et al.* [13] presented a occlusion-aware 3DMM. Their method firstly segmented the facial image into face and non-face regions, and then modelled them separately. Xiao *et al.* [14] proposed a facial landmark detection method based on the blendshapes. They utilised the expressional blendshapes to estimate the landmarks from the faces with different expressions. Other statistic model-based researches for 3D facial applications have also been proposed. However, due to the principle that the statistic models are based on, the performance of those methods highly depends on the quality and generality of the pre-trained 3D face basis components. Large errors between the modelled face and target face can occur if high variations exist from the inputs.

Non-rigid registration-based solutions. To achieve the personalised 3D facial model, the non-rigid mesh registration based methods become popular. This type of method dynamically deforms a 3D facial template towards a 3D target face input by the user. Each 3D vertex in the template is individually assigned with a deformation parameter, which normally consists of a rotation and a translation matrix. The whole process is a non-linear optimisation process.

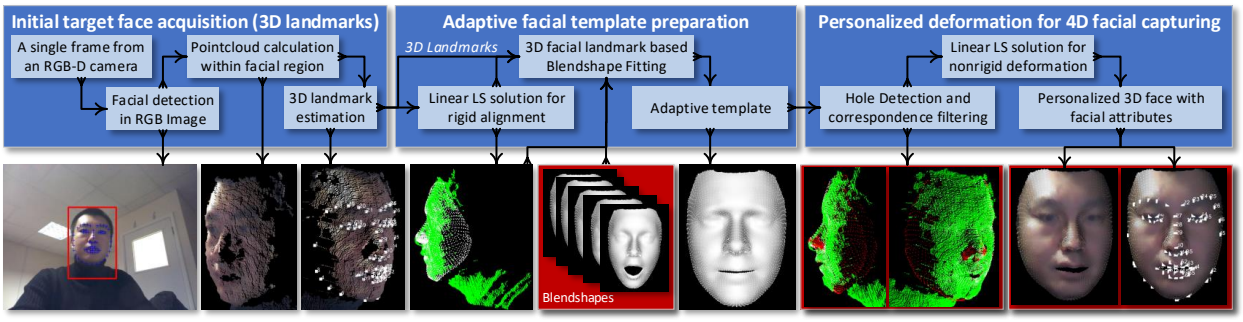

Figure 1: An illustration of the pipeline for the proposed method.

It iteratively decreases the distance between the template and the target face. For example, Ichim *et al.* [15] presented a dynamic 3D avatar creation using non-rigid facial template registration. The target face in their method was pre-acquired using structure from motion algorithm with multiple static facial images from the same expression. Then the template was iteratively deformed according to this target face. Cheng *et al.* [16] presented a non-rigid ICP method for 3D face alignment. Trimech *et al.* [17] presented a 3D facial expression recognition method. They utilised the non-rigid facial registration to achieve their recognition task. Savran *et al.* [18] also proposed a non-rigid registration based method for model-free 3D facial expression analysis. Sánta *et al.* [19] presented a non-rigid deformation method for 3D face alignment without correspondence. Other similar studies also exist for 3D facial modelling. However, those non-rigid deformation based methods may suffer from a low performance if the input target face has a low quality, for instance, incomplete facial surface, or noisy data. Additionally, there were also researches discussed the deformation methods for real-time 3D facial reconstructions based on good quality RGB-D inputs [20, 21, 22]. However, their methods achieved real-time processing by relying on GPU computation, which would also increase hardware costs.

Dedicated 3D equipment-based solutions. High performance for 3D facial modelling can be achieved if high-end or special equipment is utilised for the reconstruction process. For example, Beeler [23] discussed a photo-realistic digital human face creation using a passive capture configuration. The equipment included multiple industrial cameras and static illuminations. Villarini *et al.* [24] discussed a 3D facial reconstruction method that used the technique of photometric stereo. They utilised the equipment that required multiple directional artificial illuminations. Commonly, those special equipment based methods can achieve high fidelity of a 3D face. However, the system cost and operation complexity are relatively high compared to other types of methods. This makes those methods less able to be popularised for civil uses.

Deep learning-based solutions. Besides those three common types of approaches for 3D facial modelling, the machine learning-based methods have also been introduced recently [25]. For example, Jackson *et al.* [26] presented a method for 3D face reconstruction from a single image. They utilised a Convolutional Neural Network (CNN) variation to achieve the task. However, accuracy was still limited. Chang *et al.* [2] presented a CNN Model named ExpNet, which regresses a 29D vector as coefficients for 3D facial expression reconstruction. However, their method required GPU computation, and could only achieve about 10 frames per second with GTX TITAN X during online processing. Tewari *et al.* [27] proposed a deep network that can learn a face identity model in both shape and appearance. Their method exported a graph-based identity facial model by parameter estimations. The video frame sequences were required for their approach for 3D facial modelling. Wu *et al.* [28] explored the deep learning-based regression for 3DMM parameter estimation from multi-view inputs. They utilised and end-to-end trainable CNN to take multiple facial images as inputs for 3DMM fitting. Ramon *et al.* [29] also presented a research to utilise mutli-view facial input for 3DMM parameter regression based on siamese neural networks. All those deep learning-based methods need a dedicated GPU to achieve 3D facial modelling. The computational cost is commonly high. Additionally, those methods also require a large amount of carefully designed training data for high performance for unseen input domain.

3. Methodology overview

In this paper, to achieve simplicity while preserving accuracy, a linearly augmented 3D facial modelling method is proposed. It only utilises a low-priced consumer-grade RGB-D camera for data capturing. The modelling process is fully automatic. Without any complex operation, the user only needs to show his/her face in front of the camera for 3D facial reconstruction. Data captured by this kind of cameras is usually noisy with missing points, for example,

Table 1
The comparisons of the proposed method with Deep Learning (DL) approaches.

	The proposed method	*Most of the DL-based approaches*
How algorithm is accomplished	Optimisaions for 3D coordinates based on 3D geometric rules.	Depth map per-pixel prediction by image convolutional feature learning.
How to work with unseen data	Direct use.	Requiring model retraining, model weights fine-tuning or transfer learning.
Type of the input data	One frame of RGBD data.	One frame of RGB data.
Real-time computation cost	Requiring only CPU computation with linear optimisations.	Requiring GPU computation for both training and testing.
How 3D Model is demonstrated	A carefully designed 3D polygon mesh with rich information not affected by input occlusions.	A point cloud that comes from a depth map, which is presented from a certain viewpoint (commonly a frontal viewpoint). This means each location (x, y) in the depth map only has one z value.
How facial semantic information is included	The 3D mesh is composed of a collection of predefined vertices and edges. The relationship between vertices are deliberately designed, where facial semantic information is stably embeded. This is helpful for facial manipulations, such as facial animation manipulation.	The facial point cloud is scattered with no maintenance for inter-vertex relationships. The facial semantic information can be transferred from input data to depth map via one-on-one pixel correspondence.
Resolution of the 3D face	Determined by the template, no matter the input data is of low/high resolution.	Determined by the input data.

data missing in the nose area as shown in the middle and right parts of Fig. 2. The proposed method can well handle the low-quality data and exports a personalised and detailed 3D facial model instantly. The computational cost is also minimised, which enables real-time 4D capturing only based on a CPU. Without any pre-processing that are commonly included in most existing methods such as personalising a blendshape model or training a deep learning model for a certain group of users, the proposed method can directly and continuously capture a 4D expressional face model for a user in real-time.

As shown in Fig. 1, the proposed method consists of three main components: *the target face pre-acquisition* (Section 4), *the adaptive facial template preparation* (Section 5), and *the personalised deformation for the final output* (Section 6). The proposed method utilises a non-rigid mesh deformation scheme to achieve a personalised 3D face for an unseen target face. This non-rigid deformation involves a process where a Source Point Cloud (SPC), namely a facial template, is iteratively deforming non-rigidly towards a Target Point Cloud (TPC). By the end of this process, the deformed SPC demonstrates a desired 3D face shape as the final result. The first component in Fig. 1 is used for preparing a valid TPC as the deformation target, while the second component provides an optimised SPC to reduce the deformation cost. The third component employs linear solutions to achieve the desired non-rigid deformation. Those three components of the proposed method will be discussed in detail in the following sections.

Rather than using a popular deep learning-based solutions, the proposed method achieves the real-time 3D facial modelling based on the 3D coordination optimisation utilising the 3D geometric constraints. A 3D facial mesh is recovered with edges connecting vertices. The mesh topology can be maintained regardless of facial expressions and facial identities. This can be very helpful for those facial manipulation tasks that utilise the 3D face obtained by the proposed method, including facial animation creation, facial palsy analysis, etc. A brief comparison of pros and cons of the proposed method with the deep learning-based ones is illustrated in Table 1, along with some other sides where the proposed method can be further enhanced in the future.

4. Initial target face pre-acquisition

The proposed method only utilises a consumer-grade RGB-D camera as the input device. For the demonstrative purpose, a Kinect is adopted in this paper. An RGB-D camera can simultaneously provide an RGB image and a depth map from a single frame of data. To enhance the generality, only the RGB-D data is utilised excluding any specific functionalities of the Kinect. Thus, any RGB-D camera can be used for the proposed method. To acquire the target

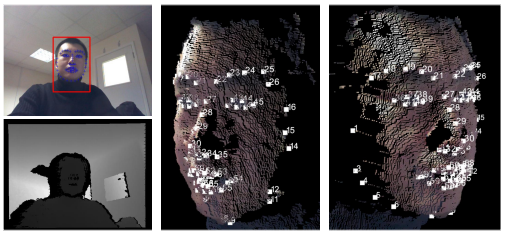

Figure 2: An illustration of the pre-acquired 3D target face with 3D facial landmarks. Large holes can be noticed in the face, especially in nose areas.

face, facial detection is firstly applied to only reserve the facial data from both the RGB image and depth map. Given the principal point and focal length of the depth camera, a point cloud can be calculated according to (1).

$$\frac{x}{u-u_0} = \frac{y}{v-v_0} = \frac{z}{f} \tag{1}$$

If (u_0, v_0) is the camera principal point, and f is the focal length, then (x, y, z) is a 3D coordinate corresponding to a 2D point (u, v) in depth map.

Human faces have a similar structure, which can be utilised to aid the 3D modelling process. In this paper, the 3D facial landmarks are extracted from the target 3D face (referred as a Target Point Cloud, TPC for short) obtained as described above. Those 3D facial landmarks will be used for the adaptive template generating, rigid pre-alignment and hole-detections. They will be discussed in the following sections. The facial landmark detection is well researched for 2D images [30, 31, 32, 33, 34], but less studied for 3D data. To simplify the 3D facial landmark detection process in the proposed method, rather than directly detecting the 3D landmarks in the TPC, we detect the 2D facial landmarks on the aligned RGB image, and find their corresponding 3D points. The 2D facial landmark detection in this paper is based on the works from Baltrusaitis *et al.* [30]. However, as shown in Fig. 2, incomplete facial points (like nose region) are expected due to the low quality of the output from a low-priced camera. Therefore, not every 2D point in the RGB image has a corresponding 3D point in the TPC. It is highly likely that the detected 2D landmark in the RGB image may have no 3D counterpart in the TPC. To tackle this, a simplified solution is adopted by finding the nearest neighbour ($k$nn, k=1) of the 2D facial landmark among the 2D points in the image, each of which has a valid 3D correspondence in the TPC. A radius constraint is applied to prevent this nearest 3D point is not too far away from its expected position. It can occur if this expected 3D facial landmark is located in a hole on the TPC. The initial target face with 3D facial landmarks is shown in Fig. 2.

5. Adaptive facial template preparation

The proposed method generates an adaptive facial template for a rough alignment to the target face before the non-rigid deformation process. Traditional non-rigid deformation-based approaches commonly utilise a fixed facial template to initiate the shape transformation. This fixed template is universal for arbitrary targets, thus introduces extra computations to achieve the target shape in the deformation process. In this paper, an adaptive template is adopted, with which the non-rigid registration process only minimises the vertex distance between two similar facial meshes. This strategy can save much computation consumption for real-time processing.

The adaptive template generation is inspired by the concept of blendshapes [35, 36], which can provide an expressional 3D face targeting for a specific person. The traditional blendshape fitting can be expressed as

$$\mathbf{f} = w_1\mathbf{b}_1 + w_2\mathbf{b}_2 + \cdots + w_n\mathbf{b}_n = \sum_i^n wi\mathbf{b}_i \tag{2}$$

Or in matrix notations as

$$\begin{cases} \mathbf{f} = \mathbf{BW} = \begin{bmatrix}\mathbf{b}_1 \ \mathbf{b}_2 \ ... \ \mathbf{b}_n\end{bmatrix} \begin{bmatrix}w_1 \ w_2 \ \cdots \ w_n\end{bmatrix}^{\mathrm{T}} \\ \mathbf{b}_i = \begin{bmatrix}x_{i1} \ y_{i1} \ z_{i1} \ x_{i2} \ y_{i2} \ z_{i2} \ \cdots \ z_{im}\end{bmatrix}^{\mathrm{T}} \end{cases} \tag{3}$$

where $\mathbf{b}_1$ to $\mathbf{b}_n$ are n facial blendshapes representing n different 3D facial expressions. They are $3m$-column vectors composed of m 3D vertices. Traditionally, they are carefully designed for a specific facial identity. $\mathbf{B}$ is an $n \times 3m$ matrix, each column of which is an individual blendshape vector. w_1 to w_n are n weights (coefficients) assigned for those blendshapes respectively. By this linear combination, a target 3D facial model can be approximated with a certain expression. In practical applications, b_1 to b_n are considered as adding vectors to a neutral expressional 3D face b_0. Therefore, (2) and (3) can be expressed as

$$\mathbf{f} = \mathbf{b}_0 + \sum_{i}^{n} w_i \mathbf{b}_i = \mathbf{b}_0 + \mathbf{BW} \tag{4}$$

According to the traditional blendshape fitting strategy, given a 2D facial image, the coefficients in $\mathbf{W}$ are estimated by iteratively re-projecting $\mathbf{f}$ onto the image and evaluating the re-projecting error regarding the original 2D face. Usually, this iterative optimisation also involves a process to estimate a camera projection matrix of the 2D image.

The proposed method targets the unconstrained 3D facial modelling, which means the facial expression of the user can be unpredictable. Unlike the most existing methods that require a user to show his/her neutral face first for pre-processing, the first expression encountered in the proposed method can be various in different scenarios. It makes the proposed method more flexible in practice. Therefore, to minimise the difference between the TPC and the deformable template for better performance, rather than using a fixed facial template, the expressional blendshape fitting is utilised to generate an adaptive 3D facial template (Source Point Cloud, SPC) for the following non-rigid deformation. The advantages of this strategy are that: (**a**) the details of the personalised 3D face are independent regarding the training/establishment of the blendshapes; (**b**) the difference between the TPC and SPC can be adaptively minimised despite the unpredictable expressions represented by the TPC; and (**c**) the adaptive template provides a better initialisation for the non-rigid registration process.

As described in the previous section, since the 3D facial landmarks are extracted automatically on the TPC, we improve the traditional blendshape fitting by employing those 3D facial landmarks. The 3D landmarks on the blendshapes can be designated when preparing those facial shapes. Therefore, a solid correspondence can be obtained per landmark between the TPC and the blendshapes. Rather than iteratively analysing the re-projection errors based on an estimated projection matrix, the blendshape fitting in the proposed method is achieved by linearly optimising a term related to the distance between the corresponding 3D facial landmarks, as described in (5).

$$\begin{cases} \arg\min\limits_{\mathbf{W}} \left|\left| \mathbf{S}_{LMK} \left(\mathbf{b}_0 + \mathbf{BW} \right) - \mathbf{l}_{TPC} \right|\right|^2 \\ \mathbf{S}_{LMK}^{\mathrm{T}} = \left[\mathbf{xi}_1 \ \mathbf{yi}_1 \ \mathbf{zi}_1 \ \mathbf{xi}_2 \ \mathbf{yi}_2 \ \mathbf{zi}_2 \cdots \mathbf{zi}_l \right]_{3b \times 3l} \\ \mathbf{i}_i = \left[0_1 \ 0_2 \cdots 1_j \cdots 0_{3b-1} \ 0_{3b} \right]_{1 \times 3b}^{\mathrm{T}} \\ \mathbf{l}_{TPC} = \left[x_1 \ y_1 \ z_1 \ x_2 \ y_2 \cdots z_l \right]_{1 \times 3l}^{\mathrm{T}} \end{cases} \tag{5}$$

where $(\mathbf{b}_0 + \mathbf{BW})$ is the same as in (4). It can be evaluated as a $3b \times 1$ column vector if there are b blendshape additions to the neutral shape $\mathbf{b}_0$. $\mathbf{l}_{TPC}$ consists of the 3D facial landmark coordinates $(x_j,\ y_j,\ z_j)$ in the TPC. If there is a total of l valid landmarks in the TPC, then $\mathbf{l}_{TPC}$ will be a $3l \times 1$ column vector. $\mathbf{S}_{LMK}$ is a $3l \times 3b$ selection matrix that stores the indices of the 3D landmarks in the blendshapes corresponding to the landmarks in the TPC. b is the total number of the vertices in a blendshape. The third row in (5) illustrates the compositions of $\mathbf{xi}_i$, $\mathbf{yi}_i$ and $\mathbf{zi}_i$, each of which is a vector with only one non-zero element (which is 1) seated at the position of i^{th} landmark in $\mathbf{b}_0$. They form the selection matrix $\mathbf{S}_{LMK}$. A linear system can be formulated accordingly as (6). By simply solving this linear least-squares problem, the blendshape coefficients stored in $\mathbf{W}$ can be obtained robustly.

$$\left(\mathbf{S}_{LMK} \mathbf{B} \right) \mathbf{W} = \mathbf{l}_{TPC} - \mathbf{S}_{LMK} \mathbf{b}_0 \tag{6}$$

Since face orientation and head pose are unpredictable in practice, a rigid pre-alignment between the blendshapes and the TPC are applied first for head pose invariance before solving the linear system of (6). The pre-alignment is acquired by solving a rigid transformation matrix between the TPC and the neutral face $\mathbf{b}_0$. This transformation matrix is then applied to all blendshapes $\mathbf{b}_i$. The facial landmarks are utilised in this process. A projection process is as

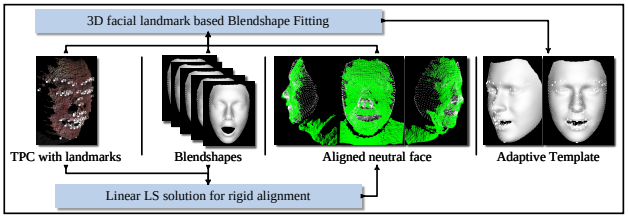

Figure 3: An illustration of the process for adaptive template generating with demonstrated results.

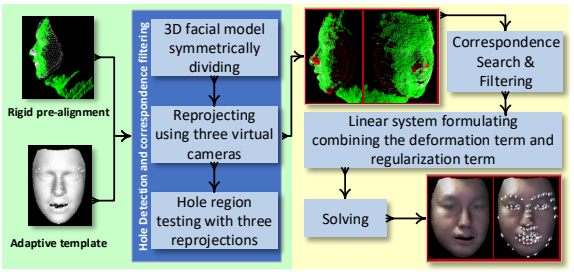

Figure 4: An illustration of the hole-detection (left) and the personalised deformation (right).

$$p' = \begin{bmatrix} x' & y' & z' & 1 \end{bmatrix}^{\mathrm{T}} = \mathbf{A}p = \begin{bmatrix} \mathbf{R} & | & \mathbf{t} \\ \mathbf{0} & | & 1 \end{bmatrix} \begin{bmatrix} x & y & z & 1 \end{bmatrix}^{\mathrm{T}} \tag{7}$$

where $\mathbf{A}$ is a transformation matrix between 3D points of p and p'. A rotation ($\mathbf{R}$) and a translation ($\mathbf{t}$) form up this transformation $\mathbf{A}$. Thus, if the 3D facial landmark from $\mathbf{b}_0$ is treated as p, and the one from TPC as p', for all landmarks,

$$\mathbf{P}\mathbf{A}^{\mathrm{T}} = \begin{bmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & z_n & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{21} & r_{31} & 0 \\ r_{12} & r_{22} & r_{32} & 0 \\ r_{13} & r_{23} & r_{33} & 0 \\ t_x & t_y & t_z & 1 \end{bmatrix} = \begin{bmatrix} x'_1 & y'_1 & z'_1 & 1 \\ x'_2 & y'_2 & z'_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x'_n & y'_n & z'_n & 1 \end{bmatrix} = \mathbf{P}' \tag{8}$$

Then, $\mathbf{A}$ can be obtained by $\mathbf{A}^{\mathrm{T}} = \mathbf{P} \backslash \mathbf{P}'$. However, in most cases, $\mathbf{A}$ not only consists of a $\mathbf{R}$ and a $\mathbf{t}$, but also a scale matrix $\mathbf{S}$, as shown in (9).

$$\mathbb{A} = \begin{bmatrix} \mathbf{S} * \mathbb{R} & | & \mathbf{t} \\ \mathbf{0} & | & 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$

To tackle this, we firstly coincide the centroid of 3D landmark set from $\mathbf{b}_0$ with the centroid of the landmark set from the TPC to achieve $[t_x,\ t_y,\ t_z] = [0,\ 0,\ 0]$. Then a SVD is conducted to decompose $\mathbf{A}$ for $\mathbf{R}$ and $\mathbf{t}$.

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}},\ \mathbf{R} = \mathbf{V}\mathbf{U}^{\mathrm{T}},\ \mathbf{t} = \begin{bmatrix} c'_x & c'_y & c'_z \end{bmatrix}^{\mathrm{T}} - \mathbf{R} \begin{bmatrix} c_x & c_y & c_z \end{bmatrix}^{\mathrm{T}} \tag{10}$$

where c_i and c'_i are the centroids of two 3D facial landmark sets on $\mathbf{b}_0$ and the TPC respectively. The calculated $\mathbf{R}$ and $\mathbf{t}$ can rigidly align all the blendshapes to the TPC. This linear alignment is much faster and more robust.

This process of the adaptive template generating is illustrated in Fig. 3 along with demonstrations of the rigid pre-alignment between the TPC and the neutral face $\mathbf{b}_0$, as well as the generated adaptive template.

6. Personalised deformation for 3D facial modelling

Many existing methods utilise the 3D Morphable Model (3DMM) or Blendshapes for 3D face reconstruction [11, 12, 37]. However, the performance of those methods highly depends on the pre-trained model. For example, fitting an Asian face using a model trained by Caucasian faces may lead to large errors. Some of the researches tackled this problem by a personalised 3DMM or blendshapes [38, 39, 40], and commonly by complex operations of pre-processing [15] for personalised blendshape preparation. In their methods, the process of the personalisation was conducted as an initialisation before the 3DMM or blendshapes were fitted to the target face. However, problems still exist because the 3DMM or blendshapes are still composed of a series of fixed 3D facial shapes no matter they are personalised or not. The errors between the target face and the modelled face are therefore still large if dramatic expressions encountered. Some expressions in the target face cannot be accurately modelled due to the high uncertainty in the input.

The proposed method tackles those problems by utilising a personalised non-rigid deformation on a pre-fitted blendshape on the target face. Rather than applying the personalisation before the fitting process, the post-deformation

can handle the uncertain inputs that might not be properly modelled by a pre-trained 3DMM or blendshapes. Moreover, the personalised deformation in the proposed method is achieved by a linear solution. It enables the proposed method achieving the result instantly. This stage of the proposed method is illustrated in Fig. 4, which is composed of two main steps: (**a**) a hole-detection for the TPC to refine the point correspondence between TPC and SPC; and (**b**) a linear system formulation to solve the personalised deformation, which non-rigidly registers the SPC to the TPC.

The point correspondence between the TPC and the SPC is crucial for the performance of the non-rigid registration. Since the SPC has already been rigidly pre-aligned to the TPC as described in the previous section, the point correspondence is achieved by the k-nearest-neighbour search ($k$nn, k=1) with the query data being each point in the SPC and the training data being all the point in the TPC. However, since the TPC is obtained using a low-priced device, low quality can be found in the TPC, which leads to large holes on the face as shown in Fig. 2. A hole-detection process will help to determine which point in the SPC may correspond to a hole region on the TPC, which means this point may have no valid corresponding point in the TPC. The hole-detection is inspired by the intuitive thought of finding holes on a 3D surface. They are obvious when viewed from certain viewpoints. The hole-detection is therefore achieved by projecting the 3D points onto multiple 2D planes from multiple viewpoints. The holes on the TPC can be then identified by 2D contouring. The projection process is described as

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{A}P = \mathbf{K}\,[\mathbf{R}|\mathbf{t}]\,P = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{11}$$

where $[x,\ y,\ 1]^{\mathrm{T}}$ is the 2D projection of a 3D point P. This projection is conducted by a virtual camera with an intrinsic parameter of $\mathbf{K}$. $[\mathbf{R}|\mathbf{t}]$ specifies the camera pose when viewing this 3D point P. In practice, the hole cannot be well perceived only from a single viewpoint if it is in a region with its normal vector perpendicular to the optical axis of viewing camera. Therefore, three virtual cameras from left, right and frontal viewpoints are utilised in the proposed method for the projections. Furthermore, only a half-face is projected when viewing from either left or right viewpoint to prevent projection overlaps, which means multiple 3D points could be projected to the same 2D position. In other words, some 3D points might be projected to a position that originally belongs to a hole, which poses challenges for hole-detection. In this paper, the facial model is divided into two symmetric halves using 3D facial landmarks. The 3D facial landmarks are roughly symmetric. It indicates that a facial middle plane can be fitted. As shown in Fig. 5, these 3D landmarks are catalogued into three groups: two for the points on the either left or right half-face, and a third group for the self-symmetric points on the nasal-ridge and in the middle of the upper and lower lips, which are supposed to be directly on the facial middle plane. By finding the mean points from the first two groups, plus the points from the third group, a roughly co-planar point set can be obtained, as shown in Fig. 5. The plane fitting is achieved by solving a linear problem. According to the definition of a 3D plane, given a roughly co-planar point set $\mathbf{P}_{\text{coplane}}$, the plane $\mathbf{A}_{\text{plane}}$ can be solved by

$$\mathbf{P}_{\text{coplane}}\mathbf{A}_{\text{plane}} = \begin{bmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & z_n & 1 \end{bmatrix} \begin{bmatrix} nx \\ ny \\ nz \\ d \end{bmatrix} = \mathbf{0} \tag{12}$$

where $(nx,\ ny,\ nz)$ is the plane normal. d is the distance of this plane to the origin of the coordinate system. Given that $(nx,\ ny,\ nz)$ is a plane normal vector, Eq. (12) can be transformed from an $\mathbf{AX} = \mathbf{0}$ problem to an $\mathbf{AX} = \mathbf{B}$ one by fixing the nx at 1.0. The plane can be then expressed as $by + cz + d = -x$, which ensures a non-trivial solution. Once solved, (nx, ny, nz) is then normalised to represent a plane normal. With the estimated facial middle plane, each point is tested using the plane equation and then assigned to the left or right face.

The three facial projections are shown in the left block of Fig. 5. The same three projections are applied for the SPC. If a projected 2D point falls into a hole in either of the three projections of TPC, it will indicate that this point has no effective correspondence. As shown in the right block of Fig. 5, the red points in the SPC have no valid point correspondence indicating they are in holes on the TPC.

Given the point correspondence between the TPC and the SPC, a linear system can be formulated to non-rigidly deform the adaptive template towards the target face for 3D facial personalisation. This optimisation process consists of two types of the terms: (**a**) the deformation term that guides the registration process towards the target shape; (**b**)

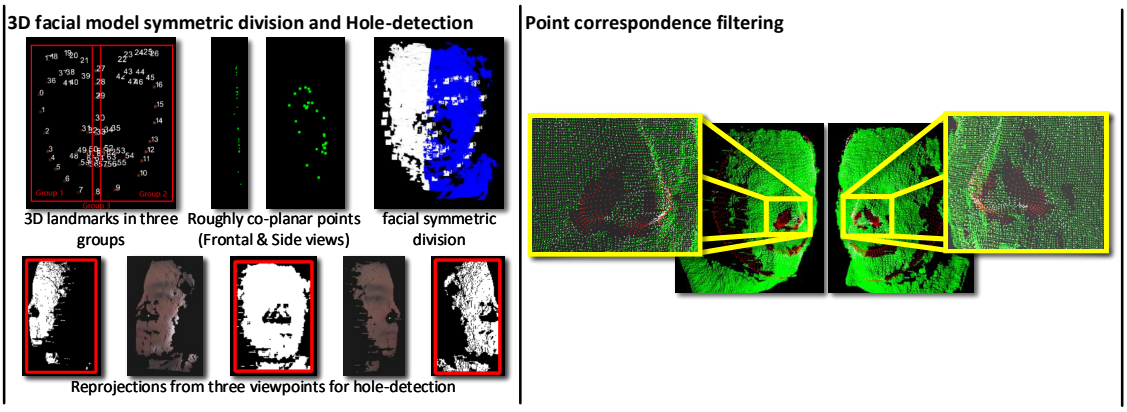

Figure 5: Illustrations of the hole-detection (left) and point correspondence filtering (right). Green points are from the TPC. Red and white points are from the SPC.

the regularisation term that enforces the topology of the SPC to make a reasonable deformation. Those two types of the terms can be formulated as in (13) and (14) respectively:

$$E_{def} = \arg\min_{T_i(\bullet)} \left(\sum_{i \in SPC_{with\_cor}} \left|\left| T_i(v_i) - corres(v_i) \right|\right|^2 \right) \tag{13}$$

$$E_{reg} = \arg\min_{T_i(\bullet)} \left(\sum_{i \in SPC} \left|\left| \left[T_i(v_i) - \sum_{j \in Onering_i} w_i \left(T_j(v_j) \right) \right] - \left[v_i - \sum_{j \in Onering_i} w_j \left(v_j \right) \right] \right|\right|^2 \right) \tag{14}$$

where $T_i(\bullet)$ denotes the transformation for the i^{th} point in the SPC. Each point has a different transformation matrix. $corres(\bullet)$ returns the corresponding point in the TPC. $Onering_i$ is the point set of the onering-neighbours of the i^{th} point in the SPC. w_j is the weight when calculating the onering centre. All onering point share the same w_j, and the sum of those weights is 1.0 for an onering. Eq. (13) states that each 3D point from the SPC that has a valid corresponding point in the TPC should move as close to its correspondence as possible. Eq. (14) states that during the deformation process, for each point in the SPC, the distance from the centre of its onering-neighbours should remain as unchanged as possible. An additional weight between those two terms is added to adjust the impact factors of those two terms to the overall optimisation, as shown in (15):

$$E_{Overall} = E_{reg} + w_{def} E_{def} \tag{15}$$

In the proposed method, we combine those two types of energy terms into a linear system for better efficiency.

According to (14), to enforce the topology of the SPC, we need to firstly calculate the regularisation constraint, which should remain as unchanged as possible during the deformation. This calculation is as shown in (16),

$$\mathbf{S}_{onering} \mathbf{V}_{SPC} = \mathbf{D}_{regularisation} \tag{16}$$

where $\mathbf{V}_{SPC}$ is an $n_{SPC} \times 3$ matrix with n_{SPC} being the number of the 3D points in the SPC. $\mathbf{V}_{SPC}$ is composed of all the coordinates $(x_i,\ y_i,\ z_i)$ of the 3D vertices from the SPC. $\mathbf{S}_{onering}$ is an $n_{SPC} \times n_{SPC}$ onering selection matrix. As shown in (17), $\mathbf{s}_i$ is i^{th} row of $\mathbf{S}_{onering}$ corresponds to the 3D vertex in the same row of $\mathbf{V}_{SPC}$. $n_{onering\_i}$ is the number of the onering-neighbouring points of the i^{th} point in the SPC. $\mathbf{D}_{regularisation}$ is a regularisation constraint, which consists of the distance between each point and its onering center.

$$\mathbf{s}_i = \left[s_{i1},\ s_{i2},\ \cdots\ s_{in_{SPC}} \right] \quad \text{, where} \quad s_{ij} = \begin{cases} 1 & \text{if } i = j \\ -1/n_{onering\_i} & \text{if } j \in \text{Onering}(i) \\ 0 & \text{Otherwise} \end{cases} \tag{17}$$

Since $\mathbf{S}_{onering}$ and $\mathbf{V}_{SPC}$ are known, $\mathbf{D}_{regularisation}$ can be calculated easily and fast. It only takes about 1 ms in our experiments. Once regularisation constraint is evaluated, a linear system that combines (13) and (14) is formulated:

$$\begin{bmatrix} \mathbf{S}_{onering} \\ w_{def}\mathbf{S}_{corres} \end{bmatrix} \mathbf{V}_{deformed\_SPC} = \begin{bmatrix} \mathbf{D}_{reg} \\ w_{def}\mathbf{V}_{corres\_in\_TPC} \end{bmatrix} \tag{18}$$

where $\mathbf{S}_{corres}$ is a $n_{corres} \times n_{SPC}$ correspondence selection matrix. n_{corres} is the number of the SPC points that have valid point correspondence in the TPC. Therefore, each row of $\mathbf{S}_{corres}$ is a selection vector to select a point that has a valid correspondence, as shown in (19). $\mathbf{V}_{deformed\_SPC}$ is a $n_{SPC} \times 3$ matrix composed of all the coordinates $(\hat{x}_i,\ \hat{y}_i,\ \hat{z}_i)$ of the 3D vertices from the deformed SPC. It is the final output of the 3D facial modelling. The vertex order (row order) of $\mathbf{V}_{deformed\_SPC}$ is the same as the one of $\mathbf{V}_{SPC}$. $\mathbf{V}_{corres\_in\_TPC}$ is a $n_{corre} \times 3$ matrix composed of the coordinates of the corresponding points in TPC. The row order of the $\mathbf{V}_{corres\_in\_TPC}$ is the same as the one of $\mathbf{S}_{corres}$. $w_{deformation}$ is a weight to adjust the balance between the two terms.

$$s_{ij} = \begin{cases} 1 & \text{If } i^{th} \text{ point in SPC is a point has valid correspondence;} \\ 0 & \text{Otherwise.} \end{cases} \tag{19}$$

Since $\mathbf{S}_{onering}$, $\mathbf{S}_{corres}$, $\mathbf{V}_{corres\_in\_TPC}$ and $\mathbf{D}_{regularisation}$ are known, the linear equation of (18) can be easily solved to estimate $\mathbf{V}_{deformed\_SPC}$ without loss of the robustness. According to the experiments, a value of 1.48 for $w_{deformation}$ can result in a well deformation performance. It only takes 6 ms to solve (18).

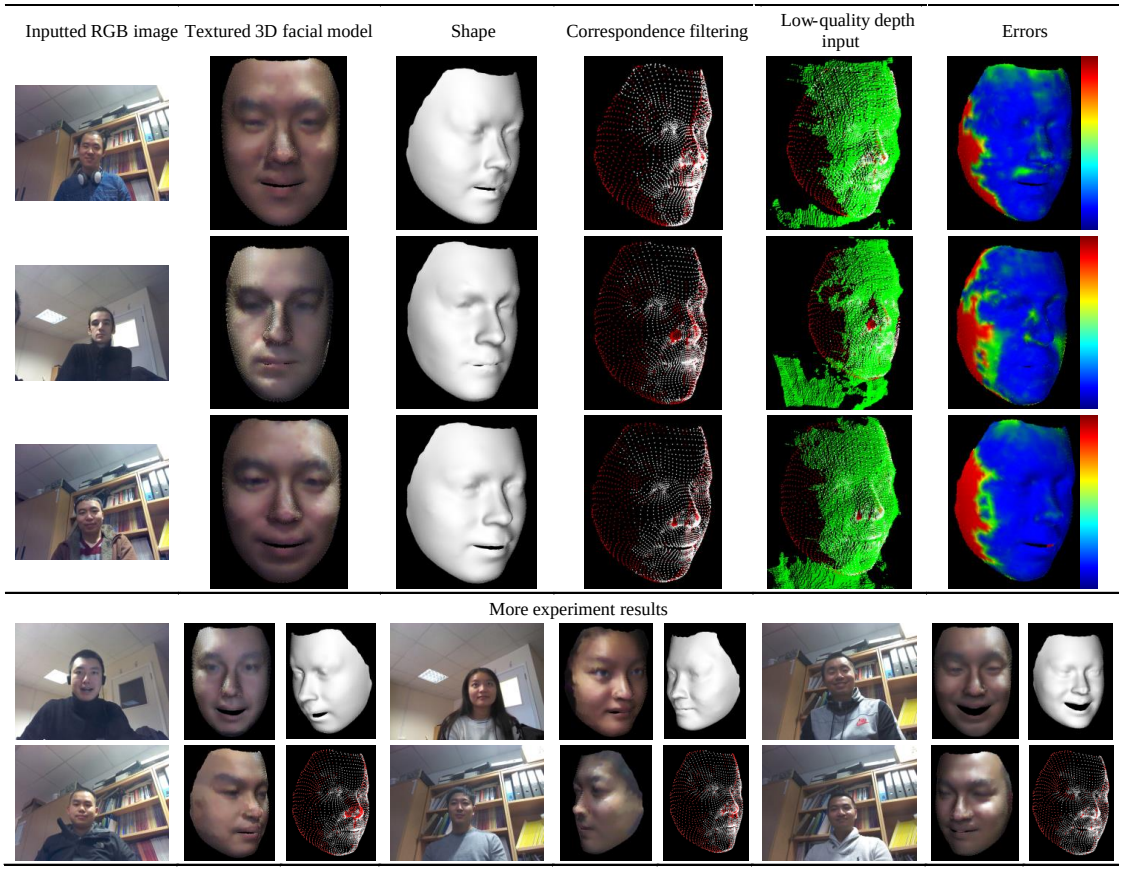

Figure 6: Samples of the experiment results. The top block demonstrates the details of the 3D facial modelling result using the proposed method. It includes (1) the captured RGB image, (2) the textured 3D facial model, (3) the 3D facial shape, (4) the 3D facial point cloud where red points have no valid correspondence in the deformation process, (5) the alignment between the input target face (green points) and the recovered 3D face (red and white points), and (6) the reconstruction error heatmap ranging from 0 mm (blue) to 5 mm (red). The bottom block demonstrates more reconstruction results.

7. Experiments and evaluations

In this section, the proposed method is evaluated using both live data from the Kinect and offline data from a public database. The experiments on the live data evaluate the performance of our method when dealing with unpredictability and low-quality in the inputs. Large holes are well handled by our method. The offline data-based experiments demonstrate the performance of the linearity augmented non-rigid deformation in our method. The experiments are conducted on a consumer-grade laptop. The proposed method is implemented only based on CPU computations.

7.1. Visual demonstrations

Experiments on live data. The Kinect is a typical consumer-grade RGB-D sensor. In this part of the experiments, a Kinect is utilised for target face acquisition. Samples of the experiment results are shown in Fig. 6. The top block of Fig. 6 demonstrates the details of the reconstruction results in six columns, which include (**1**) an inputted RGB image, (**2**) a captured 3D facial model with facial texture, (**3**) a rendered facial shape without texture, (**4**) a recovered 3D facial point cloud with red points being the vertices corresponding to the holes on the TPC, (**5**) the alignment between the TPC (green points) and deformed SPC (red and white points with red points corresponding to holes), and (**6**) the reconstruction errors. As shown in the fifth column of the top block in Fig. 6, the TPC suffers from severe loss of the data. Large holes can be experienced, especially around the nose. The proposed method can accurately identify the holes and avoid them for the correspondence assignment. The high performance can still be achieved in the reconstructed results despite such low-quality inputs are given. Dynamic expressions are also well captured. The errors of the captures are demonstrated using heatmaps in the last column of the top block in Fig. 6. The heatmap ranges from 0 mm (blue) to 5 mm (red). Most of those points with large errors (red in the heatmap) are corresponding to the holes from the TPC. Almost all the points with valid correspondence have very small errors (blue in the heatmap).

Experiments on offline data. In this part of the experiments, BU4DFE [48] database is utilised to provide the target faces for the proposed method. Those target faces have complete facial surface and low noise. Accordingly, the deformation performance of our method can be evaluated in an isolated way without the considerations of dealing with low-quality noise. As in Fig. 7, samples of the experiments are illustrated with reconstruction errors demonstrated, which are almost zeros. The 3D vertices in the deformed SPC are accurately aligned to the surface of the TPC.

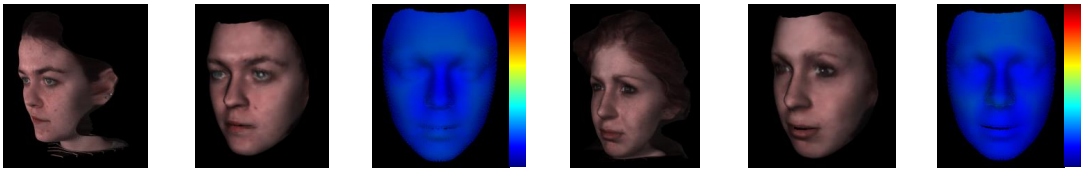

Figure 7: Samples of the experiment results using the target faces from the public database. Those target faces have a complete surface and minimal noise. The third and sixth columns demonstrate the modelling errors using the same heatmap described in Fig. 6.

Table 2
Performance comparisons in terms of Average Mean Shape Error (MSE) over all tests.

Approach	MSE (mm)	Approach	MSE (mm)
Qu *et al.* [41] (2015)	5.47	Aldrian *et al.* [42] (2010)	4.50
Liu *et al.* [43] (2019)	2.42	Kim *et al.* [44] (2018)	2.33
Ramon *et al.* [29] (2019)	2.23	Hernandez *et al.* [45] (2017)	1.92
Jiang *et al.* [1] (2018)	1.75	Sanyal *et al.* [46] (2019)	1.68
Tran *et al.* [47] (2017)	1.53	Wu *et al.* [28] (2019)	1.22
Ours [1]	**3.59**	**Ours** [2]	**1.06**

[1] The average MSE from all the points from the outputs.
[2] The average MSE only from the points that have valid correspondence.

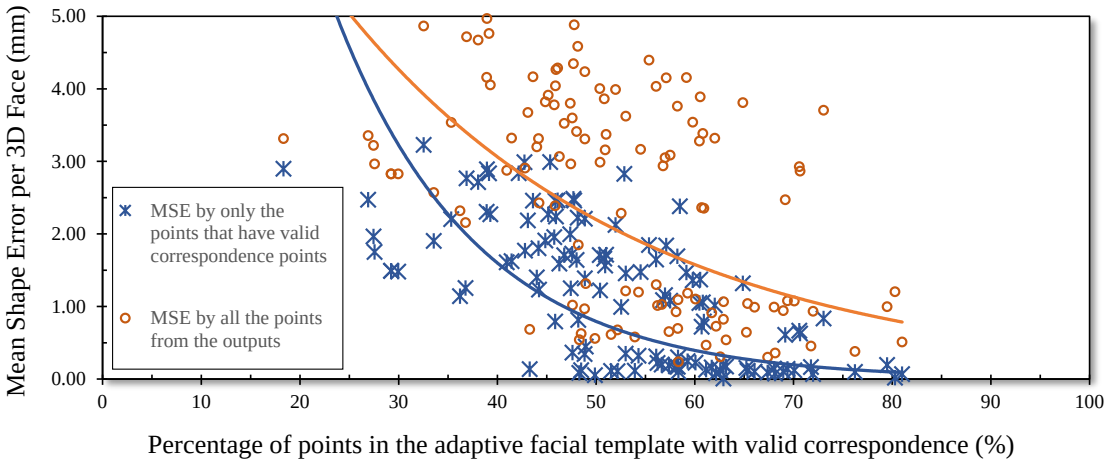

Figure 8: More experiment results by the proposed method in terms of MSE and point correspondence filtering. The blue (cross) markers are calculated only by the points that have valid correspondence, while the orange (circle) markers are calculated by all the points from the outputs. The blue and orange curved lines are fitted by the blue and orange markers respectively, indicating the trends of the 3D face capturing errors over the numbers of the effective point correspondence.

Table 3
Overall algorithm OP/S performance in terms of FLOPS[1] and INTOPS[2].

	CPU Utilisation with Vectorisation	Hardware Peak (OP/S under highest workload)[3]
FLOPS	<0.01%	254,228 (Double-Precision) FLOPS
	<0.01%	509,189 (Single-Precision) FLOPS
INTOPS	<0.03%	175,223 (Int64) INTOPS
	<0.02%	352,924 (Int32) INTOPS

[1] FLOPS (Floating-Point Operation per Second).
[2] INTOPS (Integer Operation per Second).
[3] Besides FLOP or INTOP, more tasks can also consume CPU, including data copy or memory allocation, etc.

7.2. Quanlitative evaluations

The proposed method has also been evaluated using several metric measurements illustrated in Table 2 and Fig. 8. Table 2 demonstrates the 3D modelling accuracy of our method compared with several existing approaches [41, 42, 43, 44, 29, 45, 1, 46, 47, 28]. The average Mean Shape Error (MSE) is used as the accuracy metric produced by a large group of experiment results. For the proposed method, it is calculated in two aspects: (1) the MSE by all the points from the reconstruction results, and (2) the MSE only by the points in the SPC that have valid point correspondence from the TPC. The proposed method outperforms other methods in accuracy. Fig. 8 demonstrates the MSEs from a large group of experiment results using the proposed method. The same two aspects as in Table 2 are demonstrated in Fig. 8. They are shown as orange and blue markers with more than a hundred tests for each aspect. As illustrated by the blue line and orange line fitted from the scattered points, the MSE declines with the increase of valid point correspondence establishment from the facial template to the target. They indicate that the correspondence filtering can significantly improve the performance of non-rigid deformation during the 3D face capture.

7.3. Discussions on computational cost

The proposed method has achieved a minimised computational cost by a series of linear solving processes. The 4D facial capturing can run at 20+ fps based on CPU computation on a computer with quad-core 2.7 GHz Intel Core i7. This CPU is capable of 216.1 GFLOPS (Giga Floating-Point Operations per Second) performance. The average CPU usage during the test for the proposed method is less than 0.10%. The RGB-D data resolution is 640×480. The following components in the proposed method are measured: adaptive template generating takes about 1 ms; hole detection and correspondence estimation averagely take about 10 ms; linearly non-rigid deformation averagely takes about 35 ms, which consumes less than a million FLOPs (Floating-point Operations) thanks to the linearity augmented

solutions; target face acquisition with 3D facial landmarks averagely takes 40 ms in a parallel thread. The target faces acquired are stored in a buffer for the modelling thread to retrieve during 4D capturing. As tested, once the initialisation process is done, the total run time for one frame is about 49 ms on average. More evaluations on computations are shown in Table 3 in terms of FLOPS and INTOPS.

8. Conclusion

This paper presents a method that can robustly capture a personalised 4D facial model in real-time only using a consumer-grade camera. The proposed method is fully automatic without the need for any manual intervention. A user only needs to show his/her face in front of a commodity camera for 4D face capture. The proposed method consists of three main stages: 1) acquisition of the initial target face with 3D facial landmarks from an RGB-D camera; 2) the adaptive template generation per face based on 3D facial landmarks extracted; and 3) a linear solution enhanced mesh deformation for the final 3D face model. The whole process is achieved with linearity augmentation. The computational cost and the system cost are very low. The proposed method can run in real-time for 4D capturing even on a low-end laptop based on only CPU computation. Experiments are conducted using both the live data from the camera and the offline data from a public database. The performance of the proposed method is demonstrated with the competitive results compared against several existing methods in a similar field.

References

[1] L. Jiang, J. Zhang, B. Deng, H. Li, L. Liu, 3D Face Reconstruction With Geometry Details From a Single Image, IEEE Transactions on Image Processing 27 (10) (2018) 4756–4770.

[2] F. Chang, A. T. Tran, T. Hassner, I. Masi, R. Nevatia, G. Medioni, ExpNet: Landmark-Free, Deep, 3D Facial Expressions, in: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), 2018, pp. 122–129.

[3] S. Zhang, H. Yu, T. Wang, L. Qi, J. Dong, H. Liu, Dense 3D facial reconstruction from a single depth image in unconstrained environment, Virtual Reality 22 (1) (2018) 37–46.

[4] R. S. Kute, V. Vyas, A. Anuse, Component-based face recognition under transfer learning for forensic applications, Information Sciences 476 (2019) 176–191.

[5] N. Farajzadeh, M. Hashemzadeh, Exemplar-based facial expression recognition, Information Sciences 460-461 (2018) 318–330.

[6] H. Yu, O. Garrod, R. Jack, P. Schyns, A framework for automatic and perceptually valid facial expression generation, Multimedia tools and applications 74 (21) (2015) 9427–9447.

[7] H. Yu, H. Liu, Regression-based facial expression optimization, IEEE Trans. Hum. Mach. Syst. 44 (3) (2014) 386–394.

[8] H. Yu, O. G. B. Garrod, P. G. Schyns, Perception-driven facial expression synthesis, Computers & Graphics 36 (3) (2012) 152–162.

[9] L. Chen, M. Zhou, W. Su, M. Wu, J. She, K. Hirota, Softmax regression based deep sparse autoencoder network for facial emotion recognition in human-robot interaction, Information Sciences 428 (2018) 49–61.

[10] N. Chinaev, A. Chigorin, I. Laptev, MobileFace: 3D Face Reconstruction with Efficient CNN Regression, in: The European Conference on Computer Vision (ECCV) Workshops, 2018.

[11] H. Dai, N. Pears, W. A. P. Smith, C. Duncan, A 3D Morphable Model of Craniofacial Shape and Texture Variation, in: The IEEE International Conference on Computer Vision (ICCV), 2017.

[12] C. Ferrari, G. Lisanti, S. Berretti, A. D. Bimbo, A Dictionary Learning based 3D Morphable Shape Model, IEEE Transactions on Multimedia PP (99) (2017) 1.

[13] B. Egger, A. Schneider, C. Blumer, A. Morel-Forster, S. Schönborn, T. Vetter, Occlusion-aware 3D Morphable Face Models, in: Proceedings of the British Machine Vision Conference (BMVC), York. UK, 2016.

[14] S. Xiao, J. Feng, L. Liu, X. Nie, W. Wang, S. Yan, A. Kassim, Recurrent 3D-2D Dual Learning for Large-Pose Facial Landmark Detection, in: The IEEE International Conference on Computer Vision (ICCV), 2017.

[15] A. E. Ichim, S. Bouaziz, M. Pauly, Dynamic 3D Avatar Creation from Hand-held Video Input, ACM Trans. Graph. 34 (4) (2015) 45:1–45:14.

[16] S. Cheng, I. Marras, S. Zafeiriou, M. Pantic, Statistical non-rigid ICP algorithm and its application to 3D face alignment, Image and Vision Computing 58 (Supplement C) (2017) 3–12.

[17] I. H. Trimech, A. Maalej, N. E. B. Amara, 3D facial expression recognition using nonrigid CPD registration method, in: Information and Digital Technologies (IDT), 2017 International Conference on, IEEE, 2017, pp. 478–481.

[18] A. Savran, B. Sankur, Non-rigid registration based model-free 3D facial expression recognition, Computer Vision and Image Understanding 162 (Supplement C) (2017) 146–165.

[19] Z. Sánta, Z. Kato, 3D Face Alignment Without Correspondences, in: European Conference on Computer Vision, Springer, 2016, pp. 521–535.

[20] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, Real-time non-rigid reconstruction using an RGB-D camera, ACM Transactions on Graphics (TOG) 33 (4) (2014) 156.

[21] E. Bondi, P. Pala, S. Berretti, A. Del Bimbo, Reconstructing High-Resolution Face Models From Kinect Depth Sequences, IEEE Transactions on Information Forensics and Security 11 (12) (2016) 2843–2853.

[22] P. Anasosalu, D. Thomas, A. Sugimoto, Compact and accurate 3-d face modeling using an rgb-d camera: Let's open the door to 3-d video conference, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2013, pp. 67–74.

[23] T. Beeler, Passive Spatiotemporal Geometry Reconstruction of Human Faces at High Fidelity, IEEE Computer Graphics and Applications 35 (3) (2015) 82–90.
[24] B. Villarini, A. Gkelias, V. Argyriou, Photometric Stereo for 3D Face Reconstruction Using Non Linear Illumination Models, in: IAPR Workshop on Multimodal Pattern Recognition of Social Signals in Human-Computer Interaction, Springer, 2016, pp. 140–152.
[25] A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis, Deep Learning for Computer Vision: A Brief Review, Computational Intelligence and Neuroscience 2018 (2018) 7068349.
[26] A. S. Jackson, A. Bulat, V. Argyriou, G. Tzimiropoulos, Large Pose 3D Face Reconstruction from a Single Image via Direct Volumetric CNN Regression, arXiv preprint arXiv:1703.07834.
[27] A. Tewari, F. Bernard, P. Garrido, G. Bharaj, M. Elgharib, H.-P. Seidel, P. Perez, M. Zollhofer, C. Theobalt, FML: Face Model Learning From Videos, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
[28] F. Wu, L. Bao, Y. Chen, Y. Ling, Y. Song, S. Li, K. N. Ngan, W. Liu, MVF-Net: Multi-View 3D Face Morphable Model Regression, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
[29] E. Ramon, J. Escur, X. Giro-i Nieto, Multi-View 3D Face Reconstruction in the Wild Using Siamese Networks, in: The IEEE International Conference on Computer Vision (ICCV) Workshops, 2019.
[30] T. Baltrušaitis, P. Robinson, L. P. Morency, Openface: an open source facial behavior analysis toolkit, in: Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on, IEEE, 2016, pp. 1–10.
[31] X. Xiong, F. De la Torre, Supervised descent method and its applications to face alignment, in: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, IEEE, 2013, pp. 532–539.
[32] F. De la Torre, W.-S. Chu, X. Xiong, F. Vicente, X. Ding, J. F. Cohn, IntraFace, in: IEEE International Conference on Automatic Face and Gesture Recognition (FG), 2015.
[33] T. F. Cootes, C. J. Taylor, D. H. Cooper, J. Graham, Active shape models-their training and application, Computer vision and image understanding 61 (1) (1995) 38–59.
[34] I. Matthews, S. Baker, Active appearance models revisited, International journal of computer vision 60 (2) (2004) 135–164.
[35] P. Huber, G. Hu, R. Tena, P. Mortazavian, P. Koppen, W. J. Christmas, M. Ratsch, J. Kittler, A multiresolution 3D morphable face model and fitting framework, in: Proceedings of the 11th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, 2016.
[36] P. Huber, G. Hu, J. R. Tena, P. Mortazavian, W. P. Koppen, W. J. Christmas, M. Rätsch, J. Kittler, A Multiresolution 3D Morphable Face Model and Fitting Framework, 2016.
[37] P. Koppen, Z.-H. Feng, J. Kittler, M. Awais, W. Christmas, X.-J. Wu, H.-F. Yin, Gaussian mixture 3D morphable face model, Pattern Recognition.
[38] J. Thies, M. Zollhöfer, M. Nießner, L. Valgaerts, M. Stamminger, C. Theobalt, Real-time expression transfer for facial reenactment, ACM Transactions on Graphics (TOG) 34 (6) (2015) 183.
[39] J. Roth, Y. Tong, X. Liu, Adaptive 3D face reconstruction from unconstrained photo collections, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4197–4206.
[40] D. Thomas, R.-I. Taniguchi, Augmented Blendshapes for Real-time Simultaneous 3D Head Modeling and Facial Motion Capture, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3299–3308.
[41] C. Qu, E. Monari, T. Schuchert, J. Beyerer, Adaptive Contour Fitting for Pose-Invariant 3D Face Shape Reconstruction, in: Proceedings of the British Machine Vision Conference (BMVC), BMVA Press, 2015, pp. 87.1–87.12.
[42] O. Aldrian, W. Smith, A Linear Approach of 3D Face Shape and Texture Recovery using a 3D Morphable Model, in: Proceedings of the British Machine Vision Conference (BMVC), BMVA Press, 2010, pp. 75.1–75.10.
[43] P. Liu, Y. Yu, Y. Zhou, S. Du, Single View 3D Face Reconstruction with Landmark Updating, in: 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), 2019, pp. 403–408.
[44] H. Kim, M. Zollhöfer, A. Tewari, J. Thies, C. Richardt, C. Theobalt, InverseFaceNet: Deep Monocular Inverse Face Rendering, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
[45] M. Hernandez, T. Hassner, J. Choi, G. Medioni, Accurate 3D face reconstruction via prior constrained structure from motion, Computers & Graphics 66 (2017) 14–22.
[46] S. Sanyal, T. Bolkart, H. Feng, M. J. Black, Learning to Regress 3D Face Shape and Expression From an Image Without 3D Supervision, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
[47] A. Tuan Tran, T. Hassner, I. Masi, G. Medioni, Regressing Robust and Discriminative 3D Morphable Models With a Very Deep Neural Network, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
[48] L. Yin, X. Chen, Y. Sun, T. Worm, M. Reale, A high-resolution 3D dynamic facial expression database, in: Automatic Face & Gesture Recognition, 2008. FG'08. 8th IEEE International Conference on, IEEE, 2008, pp. 1–6.