3D Facial Performance Capture from Monocular RGB Video

SHUANG LIU

A thesis submitted in partial fulfilment of the requirements of

Bournemouth University for the degree of

Doctor of Philosophy

May, 2017

Abstract

3D facial performance capture is an essential technique for animation production in featured films, video gaming, human computer interaction, VR/AR asset creation and digital heritage, which all have huge impact on our daily life. Traditionally, dedicated hardware such as depth sensors, laser scanners and camera arrays have been developed to acquire depth information for such purpose. However, such sophisticated instruments can only be operated by trained professionals. In recent years, the wide spread availability of mobile devices, and the increased interest of casual untrained users in applications such as image, video editing, virtual and facial model creation, have sparked interest in 3D facial reconstruction from 2D RGB input.

Due to the depth ambiguity and facial appearance variation, 3D facial performance capture and modelling from 2D images are inherently ill-posed problems. However, with strong prior knowledge of the human face, it is possible to accurately infer the true 3D facial shape and performance from multiple observations captured with different viewing angles. Various 3D from 2D methods have been proposed and proven to work well in controlled environments. Nevertheless there are still many unexplored issues in uncontrolled in-the-wild environments. In order to achieve the same level of performance in controlled environments, interfering factors in uncontrolled environments such as varying illumination, partial occlusion and facial variation not captured by prior knowledge would require the development of new techniques.

This thesis addresses existing challenges and proposes novel methods involving 2D landmark detection, 3D facial reconstruction and 3D performance tracking, which are validated through theoretical research and experimental studies.

3D facial performance tracking is a multidisciplinary problem involving many areas such as computer vision, computer graphics and machine learning. To deal with the large variations within a single image, we present new machine learning techniques for facial landmark detection based on our observation of the facial features in challenging scenarios to increase the robustness. To take advantage of the evidence aggregated from multiple observations, we present new robust and efficient optimisation techniques that impose consistency constrains that help filter out outliers. To exploit the person-specific model generation, temporal and spatial coherence in continuous video input, we present new methods to improve the performance via optimisation.

In order to track the 3D facial performance, the fundamental prerequisite for good results is the accurate underlying 3D model of the actor. In this thesis, we present new methods that are targeted at 3D facial geometry reconstruction, which are more efficient than existing generic 3D geometry reconstruction methods.

Evaluation and validation were obtained and analysed from substantial experiment, which shows the proposed methods in this thesis outperform the state-of-the-art methods and enable us to generate high quality results with

less constraints.

Acknowledgements

I would like to express my deepest gratitude to my supervisors Prof. Xiaosong Yang and Prof. Jian Jun Zhang for their guidance and support throughout the journey of my doctorate study.

I would like to thank all the Bournemouth university supporting staff who helped to facilitate our study and research. I would also like to give special thanks to the Graduate School for the studentship.

Finally, I would like to thank my family and friends for their love and support.

Declaration

This thesis has been created by myself and has not been submitted in any previous application for any degree. The work in this thesis has been undertaken by myself except where otherwise stated.

Related Publications

- **Liu, S.**, Wang, Z., Yang, X. and Zhang, J., 2017. Realtime Dynamic 3D Facial Reconstruction for Monocular Video In-The-Wild. In Proceedings of the International Conference on Computer Vision Workshops (pp. 777-785).
- **Liu, S.**, Zhang, Y., Yang, X., Shi, D. and Zhang, J.J., 2017. Robust facial landmark detection and tracking across poses and expressions for in-the-wild monocular video. Computational Visual Media, 3(1), pp.33-47.
- **Liu, S.**, Yang, X., Wang, Z., Xiao, Z. and Zhang, J., 2016. Realtime facial expression transfer with single video camera. Computer Animation and Virtual Worlds, 27(3-4), pp.301-310.
- Zhang, Y., **Liu, S.**, Yang, X., Zhang, J. and Shi, D., 2017. Supervised coordinate descent method with a 3D bilinear model for face alignment and tracking. Computer Animation and Virtual Worlds, 28(3-4).
- Zhang, Y., **Liu, S.**, Yang, X., Shi, D. and Zhang, J.J., 2016. Sign-Correlation Partition Based on Global Supervised Descent Method for Face Alignment. In Asian Conference on Computer Vision (pp. 281-295). Springer.
- Hu, W., Wang, Z., **Liu, S.**, Yang, X., Yu, G. and Zhang, J.J., 2017. Motion Capture Data Completion via Truncated Nuclear Norm Regularization. IEEE Signal Processing Letters.
- Jiang, T., Qian, K., **Liu, S.**, Wang, J., Yang, X. and Zhang, J., 2017. Consistent as-similar-as-possible non-isometric surface registration. The Visual Computer, pp.1-11.
- Wang, Z., **Liu, S.**, Qian, R., Jiang, T., Yang, X. and Zhang, J.J., 2016, November. Human motion data refinement unitizing structural sparsity and spatial-temporal information. In Signal Processing (ICSP), 2016 IEEE 13th International Conference on (pp. 975-982). IEEE.

- Wang, Z., Feng, Y., **Liu, S.**, Xiao, J., Yang, X. and Zhang, J.J., 2016, May. A 3D human motion refinement method based on sparse motion bases selection. In Proceedings of the 29th International Conference on Computer Animation and Social Agents (pp. 53-60). ACM.

Related Projects

- **TruFace** Markless Realtime Facial Performance Tracking
  Disney Research and **IL&M**
- **DigTrace** 3D Footprint Reconstruction from 2D Images
  Bournemouth University
- **NeurAvatar** Intelligent Neurological Disorder Diagnostic System
  Bournemouth University

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Facial motion capture is the process of electronically converting the movements of a person's face into a digital database using cameras or laser scanners (Williams 1990). It automatically captures the pose, expression and facial geometry of the target actor by using the techniques of computer vision, machine learning and computer graphics. 3D facial performance tracking has been an active research topic for a long time, because of its wide applications in media production such as film, games, TV, digital heritage, VR/AR and teleconference, etc. However, with the emergence of new hardware such as smart phones with incredible computing powers and new technologies such as Virtual Reality (VR) (Li et al. 2015) and Augmented Reality (AR), comes the new demands and requirements that traditional 3D facial performance tracking methods cannot deal with from the perspectives of efficiency, accuracy, realism and responsiveness.

There are two categories of 3D facial performance tracking methods: **marker** based and **markerless** based. The **marker** based approach is invasive and requires special setup, therefore it is not suitable many casual applications. This thesis focuses on the **markerless** approach as it is more accessible, flexible and user-friendly for non-technical users. The overall pipeline of 3D facial performance tracking is shown in Fig. 1.1.

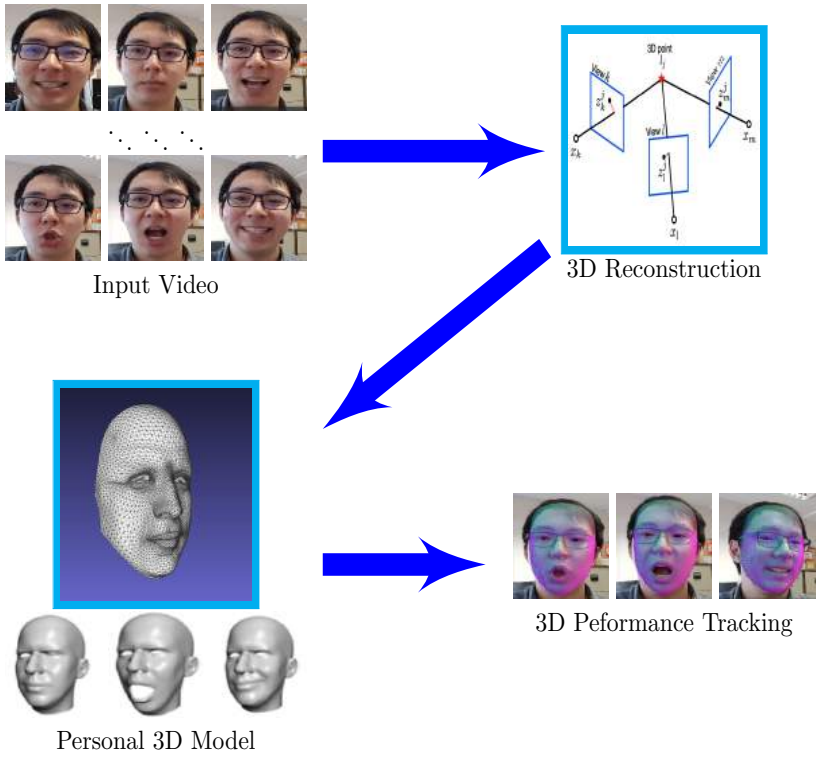


Figure 1.1: *Given an image sequence, facial landmarks are first detected, then a person specific 3D facial model is computed from solving the nonrigid 3D to 2D alignment problem. Finally 3D facial performance is tracked by aligning the 3D person specific model to the observed image.*

1.1 Background

3D facial performance capturing has many applications in gaming, film production, VR/AR interaction, teleconferencing and digital heritage, where the facial performance of a person can be processed digitally and used to drive the performance of virtual characters and creating special visual effects. With the wide spread availability of mobile devices equipped with cameras, the number of facial photos is ever increasing and there is a growing demand to reconstruct 3D facial geometry and capture performance from 2D images. For technology like 3D facial mesh tracking, it used to be only available to high production value studios such as Disney, Industrial Light& Magic, Pixar and Dreamworks, where expensive and specialised hardware have been developed to capture the performance with highest accuracy possible.

There are three types of 3D facial reconstruction and performance capturing available. First type of method is **marker** based motion capture system (MOCAP). They provide very accurate tracking but are not portable or flexible. These systems can only provide sparse tracking results and require the user to wear invasive markers. Dedicated software and trained personnel are needed to operate MOCAP, and the system may have specific requirements for the **controlled** environment. Second type of method is **depth** based. Laser scan and light field devices are very accurate but expensive and also require a lot of operational effort. Although recently consumer grade depth sensor (such as Kinect) has made its way to the market and achieved certain level of success, a large body of work have been done utilizing these devices. Nevertheless, RGB-D images and videos are still rare and not equally available as RGB images and videos. As these sensors utilize an infrared projector and camera, they typically have a practical ranging limit of 1.2 to 3.5m distance and can be only used in indoor environment where there is little to no ambient infrared. Third type of methods operate purely on **RGB** images captured in **uncontrolled** settings. As the depth information is not available, they use visual cues to infer the depth of the image scene. Usually multiple images from different poses are needed, unless depth prior of the scene is available.

Image based methods are very flexible and efficient. Although some of them are designed for controlled lab environment but recently there has emerged a increasing number of research in uncontrolled settings.

This is because they require less hardware and have a broader range of applications. Previously most 2D and 3D based methods are mostly designed for controlled lab environment, where lighting, camera calibration and facial movement are tightly constrained. In uncontrolled environment, for example online videos, historic footages and outdoor images with changing lighting condition and background, 3D facial reconstruction and performance capturing are more challenging than controlled environment due to the large variation. The ability to reconstruct and capture 3D facial geometry and performance in uncontrolled settings could enable a large range of application on images and videos captured on hand held devices. Note that in terms of 2D landmark detection, the uncontrolled settings is often refereed as in-the-wild, this report will borrow this terminology to refer to the facial performance capture in uncontrolled settings.

Many algorithms have been introduced to deal with its intrinsic difficulty caused by depth ambiguity of 2D images, lighting and camera setup variation. However, due to the nonrigid nature of human's faces and the lighting, perspective and expression variation, traditional methods cannot be directly applied to this problem, and sometimes it require manual effort for initialization. Hence, more work need to be carried out to relax the constraints of camera calibration, lighting control and increase the robustness.

This research aims to tackle the challenges of facial geometry **reconstruction** and performance **capturing** from images and videos in **uncontrolled setting**. The output of this research will help computer to better understand facial expression, recognize, track and register faces from in-the-wild images. The potential application of the developed technology will have huge impact on social life in areas such as film production, gaming, VR/AR interaction, teleconferencing and digital heritage. For instance, it will allow us to easily building virtual 3D character for gaming, expression editing and per-

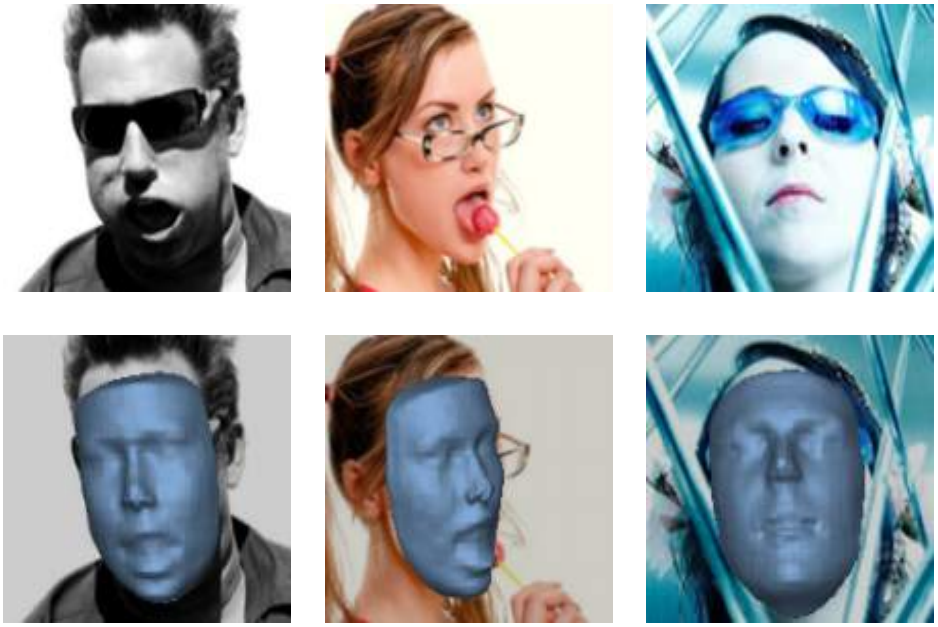


Figure 1.2: *Faces captured in uncontrolled settings exhibit drastically different appearances and poses, and the visible facial areas are often occluded by external objects.*

formance cloning. And in terms of teleconferencing to be more specifically, it will enable next generation virtual video calls driven by realistic avatars, which can also be 3D printed for personalized action figurine.

1.2 Main Challenges

The crucial part of 3D facial performance tracking are the accuracy of reconstructed facial geometry and tracked facial deformation. Typically, the facial geometry of the actor has to be reconstructed firstly from the available imagery. Unlike traditional structure-from-motion problems, under uncontrolled setting the pose and expression of the actor, and the lighting condition and camera placement are all unknown, thus making the problem ill-posed. Fortunately, previous works have exploited and developed strong prior of the facial geometry and appearance, which adds necessary regularization and constraint to the problem thus making it possible to be solved.

The main challenges for image-based 3D facial reconstruction includes:

- **Appearance Variation** Facial appearance variation comes from intrin-

sic factors such as face variability between individuals and also due to extrinsic factors such as partial occlusion, illumination, expression, pose and camera resolution. As illustrated in Fig. 1.2, facial landmarks can sometimes be only partially observed due to occlusions of hair, hand movements or self-occlusion due to extensive head rotations.

- **Acquisition Conditions** Acquisition conditions of the training and the testing data, such as environment ambient and specular lighting, resolution, background clutter can affect the landmark localization performance. This is attested by the fact that landmark localizers trained in one database have usually inferior performance when tested on another database. Moreover, certain approaches that work well in controlled or person specific settings do not work in uncontrolled or person generic settings.

- **Depth Ambiguity** Due to the varying pose and expression of the faces, the monocular camera setup and the uncontrolled capture environment, traditional stereo vision methods could not be directly applied.

- **Correspondence Matching** Establishing 3D correspondences between subsequent image-based face reconstructions is also a major challenge. Since both the pose and nonrigid deformation of the face are unknown, it is an ill-posed problem thus requires carefully designed prior to work.

1.3 Research Aims

The aim of this research is to solve these key technical challenges in facial geometry reconstruction and performance capture from 2D images and videos. The major tasks range from 2D landmark detection, coarse 3D model estimation and dense 3D model estimation. This research will propose a completely automated end-to-end method for reconstructing and tracking facial geometry and performance from images and videos. Techniques developed in this thesis could be applied in 3D character creation, 3D facial recognition and registration, 3D facial editing, which are useful for huge impact applications

such as film production, gaming, VR/AR interaction, teleconferencing and digital heritage.

1.4 Research Objectives

In order to achieve the above mentioned aim, following objectives need to be accomplished:

- **Literature Review**: review and investigate current researches on 2D landmark detection, 3D facial mesh reconstruction and tracking. Identifying the limitation of current approaches in in-the-wild situations.
- **2D Facial Landmark Detection**: design and develop a novel 2D facial landmark tracker that is able to predict the position of facial landmarks from in-the-wild images or videos.
- **3D Facial Mesh Tracking**: design and develop a novel method for tracking 3D facial performance from in-the-wild videos.
- **3D Facial Geometry Reconstruction**: design and develop a novel method for reconstructing 3D facial meshes from in-the-wild image and videos.

1.5 Contribution

There are several major contribution in this research.

- Propose and develop the novel sign-correlation training samples partition scheme that improves the performance of the supervised descent method for facial landmark detection for faces across large pose variations.
- Propose and develop a new method with a 3D bilinear model for directly learning to predict the facial landmarks with a 3D representation, which outperforms 2D based methods.

- Propose a novel method for transferring expression between source and target actors in 3D.
- Propose a novel method that corrects the 3D tracking results solved from sparse 2D facial landmarks via dense per-pixel optical flow correction.
- Propose a novel method that tracks and reconstruct the facial geometry in monocular videos in realtime via direct optimisation on GPU.

1.6 Scope and Limitation

This work focuses on 3D facial performance capture and 3D facial geometry reconstruction from monocular RGB input. Due to the inherent lacking of depth information, 3D information is inferred from the observation of the same face under different view perspectives. As a result, the performance and accuracy are inferior to those that aim for capturing ultra fine details in controlled studio setting and with specialised hardware. However, the new methods developed in this work require only ordinary consumer grade cameras and can offer more flexibility in terms of environment, lighting and facial movements.

1.7 Structure of the following chapters

- **Chapter 2** Literature review on the related research topics, including landmark tracker, 3D facial meshing tracking and 3D reconstruction techniques.
- **Chapter 3** Analysis of classic facial landmark tracking methods and introduce a new algorithm based on sign correlation which improves the landmark prediction accuracy.
- **Chapter 4** presents a novel method for 2D facial landmark detection in uncontrolled environments, and compare to state-of-the-art 2D facial

landmark detector methods.

- **Chapter 5** presents a lightweight realtime 3D facial mesh tracking method based on the 2D landmarks from Chapter 4, and an efficient algorithm to transfer expression from source actor to target user based on the tracked 3D mesh. Based on the 3D facial tracking results from the sparse 2D landmarks, a robust offline method is also presented, which automatically corrects the tracked results via optimising photometric term computed from 2D optical flow.

- **Chapter 6** presents a realtime method that densely track the performance of all visible facial area on a pixel-wise level, and dynamically reconstruct dense per-pixel facial geometry from uncontrolled video with unconstrained facial performance.

- **Chapter 7** Future plan.

Chapter 2

Related Work

In this chapter, the related works on the key techniques including 2D facial landmark tracking, 3D performance capture and mesh reconstruction will be reviewed.

2.1 2D Facial Landmark Detection

Facial landmark detection has important application in a lot of areas (Wang et al. 2014), for example biometric-based recognition (Shi et al. 2006), image editing (Yang et al. 2011), facial performance capture (Hsieh et al. 2015). As shown in Fig. 2.1, these landmarks represent a complete set of basic facial actions, and therefore allow the representation of most natural facial expressions. Exaggerated values permit the definition of actions that are normally not possible for humans, but could be desirable for cartoon-like characters. Nevertheless, various landmark markup schemes have been proposed as illustrated in Fig. 2.2, because different projects and applications might have goals other than facial animation.

In the standard approaches for 2D facial landmark detection, face bounding box detection is applied independently at each frame of the video followed by facial landmark localisation. Therefore, the performance of landmark detection depends heavily on the bounding box. In this work, experiments are

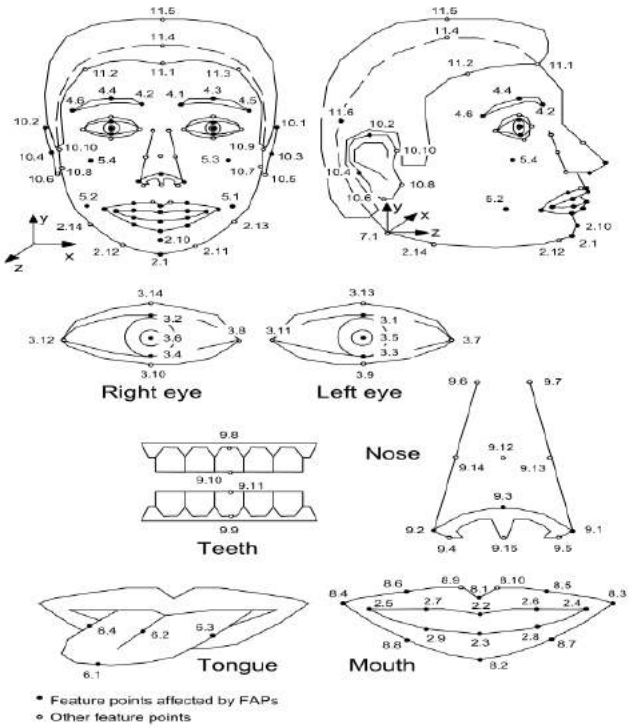


Figure 2.1: *The MPEG-4 FBA landmark markup scheme (Pandzic and Forchheimer 2003) that is commonly used in generating facial animation, which is based on the study of minimal facial actions and are closely related to muscle actions.*

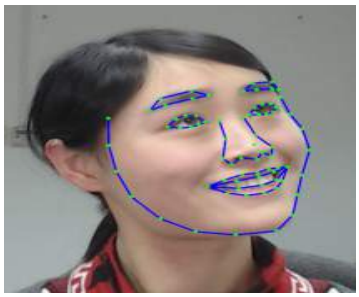
conducted with various different combination of facial bounding box detection method and facial landmark detector to find the best balance between quality and efficiency. Additionally, a novel landmark detection method is proposed and compared against state-of-the-art methods.

A multitude of works have been published to address this task. Although state of the art methods have achieved impressive performance (Kazemi and Sullivan 2014; Xiong and De la Torre 2015; Burgos-Artizzu et al. 2013; Cao et al. 2014c; Ren et al. 2014; Zhu et al. 2015), due to the large variation caused by different lighting condition, view point and partial occlusion. Many researchers have done a great deal of effort in dealing with these variations. Existing methods can be categorized into the following four groups:

- Constrained local model based;
- Active appearance model based;
- Regression-based;

(a) *68*

(b) *194*



(c) *74*

(d) *84 (2.5D)*

Figure 2.2: *There exists various landmark markup schemes. For example the 2D 68 points scheme (Sagonas et al. 2013a) in Fig. 2.2a, the 2D 194 points scheme (Le et al. 2012a) in Fig. 2.2b, the 2D 74 points scheme (Cao et al. 2014a) in Fig. 2.2c and the pseudo-3D 84 points scheme (Zafeiriou et al. 2017) in Fig. 2.2d. As shown in the figures, in different markup schemes, both the semantic meaning and the numbers of facial landmarks vary. The 2D markup schemes aligns to the 2D features observed in the image whereas the 3D markup scheme aligns to 3D features that might not be directly observable in the image.*

- Hybrid methods.

2.1.1 Constrained local model

Constrained local model (CLM) consists of shape model and group of local experts that are trained to detect independent facial landmarks. The overview of CLM is illustrated in Fig. 2.3. For a specific landmark, its surrounding appearance variance is limited. These methods have independent experts that model these variance individually. Given an image and the constrained local image patch where each experts should be in, they produce a response map of where the facial landmarks are most likely to be. The exact location of the facial points can then be inferred from these response maps and the shape model prior (Cootes and Taylor 1992). By sampling from training images,

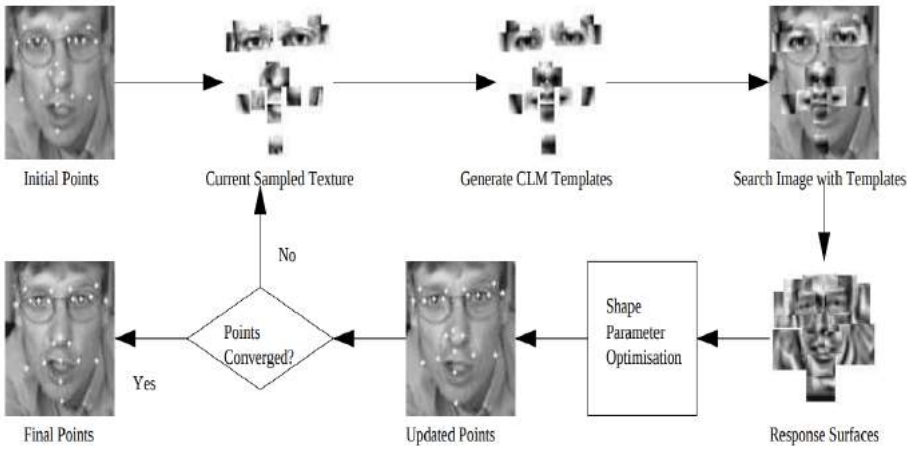


Figure 2.3: *The CLM (Cootes and Taylor 1992) search algorithm fits the joint model to the current set of feature points to generate a set of templates, then use the shape constrained search method to predict a new set of feature points. The whole process is repeated until converge.*

a multivariate Gaussian distribution can be learned, which is also known as point-distribution-model (PDM). This model is usually approximated by a principle component analysis (PCA) (Jolliffe 2002) and compressed by truncating low eigenvalue vectors. To improve the performance of the local experts and the accuracy of the classifier, image patch based Mahalanobis distance (Zhu and Ramanan 2012), support vector machines (Wang et al. 2008) and regressors (Cristinacce and Cootes 2007) have been used. This group of methods are the first successful facial landmark detection methods. However, due to the limited expressiveness and discriminative power of their local experts and the point distribution model, the performance of these models have been surpassed by other methods.

An alternative way of building the local experts is to exploit regressors instead of classifiers. In the works of Friedman et al. (2000); Cristinacce and Cootes (2007) learn a regressor from the local neighborhood appearance to the displacement between its center and the true facial feature point location. A fixed mapping function, i.e. regressor, would take complex form to incorporate the computational and generalization of facial information, Saragih et al. (2009) introduced a bilinear model to adopt face variation in pose, identity, expression and illumination. Cootes et al. (2012) later proposed to learn a random forest (Breiman 2001) that takes Haar-like feature (Lienhart and

Maydt 2002) as input. Similar patch based features such as histogram of oriented gradient (HOG) (Dalal and Triggs 2005) , SIFT (Lowe 2004) are more robust and stable in noisy images than pixel intensity difference based features (Markuš et al. 2013) albeit slower. The point distribution model statistically models the shape and regularizes the global shape configuration, which provide more information than classifiers, such as the distance of negative patches from a positive patch, where classifiers only output whether an image path is positive or negative. To sum up, these methods exploit the texture information of the local region around facial landmarks, their computational cost is usually higher than later introduced regressor methods and their performance is inferior.

2.1.2 Active appearance model

Active appearance model (AAM), as shown in Fig. 2.4, models texture of the face and detects facial shapes by trying to minimize the texture differences between training images and testing images. They contrasts CLM methods in that they model the appearance variation from a holistic point of view, both of which are generally represented by linear combinations (Cootes et al. 2001). The shape model can be constructed in similar ways as CLM-based methods. The texture model is constructed from normalized mean shape training images, which is then compressed using PCA. In testing time a synthesized model is first generated then its discrepancy to the real image is minimized via optimization method. It is a simple and intuitive model, however its performance in real world applications suffers from low efficiency, discriminative power and robustness in uncontrolled environment.

In terms of the texture model, not surprisingly, person-specific AAM is easier to build than person generic AAM due to the large variation (Gross et al. 2005). To overcome the significant appearance variation, Papandreou and Maragos (2008) proposed to incorporate prior information in the AAM mean template update to constraint the fitting process. Additive boosting models (Tresadern et al. 2010) have also been exploited to alleviate the slow

(a) **(b)** **(c)**

Figure 2.4: *The AAM (Cootes et al. 2001) algorithm uses the difference between the current estimate of appearance and the target image to drive an optimization process. By taking advantage of the least squares techniques, it can match to new images very swiftly. Starting from Fig. 2.4a, the average face is initialized and compared to the observed image, the difference between synthesized face and real image is computed as in Fig. 2.4b, and the face template is adjusted accordingly, the whole process is iterated until it converge to a good fit as shown in Fig. 2.4c.*

convergence in the original (Cootes et al. 1998) linear regression. Because the linear regression strategy is a coarse approximation of the nonlinear relation between texture residuals and warp parameters, Saragih and Goecke (2007) introduced a nonlinear boosting procedure to learn the multivariate regression.

The nonlinear relationship can be learned by boosting, which construct an ensemble of weak classifiers, which take Haar-like rectangular features as input and output shape updates. The learning procedure finds a set of shape updates that maximize the score of the strong classifier consists of these weak classifiers. However, since these weak classifiers are used to classify the point distribution model parameters, it cannot guarantee the fitting objective will converge to the optimum solution. To solve it, a classier can be learned to determine whether or not to switch from one shape parameter to another one (Wu et al. 2008).

Both ASM and AAM model the shape variation by certain parameters such as those in point distribution model. They both seek to minimize the distance between its learned model and the seen image. The key difference between them is that ASM model minimize and search for possible key point location using texture information around the key points specified by the user

whereas AAM model minimize the distance between the synthesized model image and the target image, thus modeling the appearance of the whole face (Cootes et al. 1999). As a result, ASM is faster and achieves more accurate feature point location than AAM. However, as it explicitly minimize texture errors AAM gives a better matching to the image texture. In summary, AAM works well in controlled environment when person specific training data is available, however it generally does not work well in uncontrolled environment.

2.1.3 Regression based methods

Regression based methods, sometimes called regressors, directly tries to predict the location of facial landmarks by learning a mapping function that takes the image as input and outputs predicted facial landmark locations (Zhou and Comaniciu 2007). An example of the regression based method pipeline is shown in Fig. 2.5. The shape and appearance are modeled inexplicably although in some literature people have achieved performance boost by introducing heuristic priors. The features could be Haar-like feature (Viola and Jones 2004), scale invariant feature transform (SIFT) (Lowe 2004) and local binary pattern (LBP) (Ojala et al. 1996). Recently these model are gaining popularity because the use of regression tress trained with boosted shape-indexed-feature which has the nice property of changing the location of extracted feature with regard to current shape, which makes them have certain level of semantic meanings, such as feature point next eye brow, mouth corner and nostril. The fact that they are pixel based instead of patch based also makes them cheaper to evaluate (Dollar et al. 2010; Cao et al. 2014c).

Recently these pixel intensity based methods have been pushing the state-of-the-art performance both in regards to accuracy and speed. In particular, they are the record holders of the highest accuracy on LFPW: labeled face parts in the wild database (Belhumeur et al. 2013), 300-W: 300 faces in the wild challenge (Sagonas et al. 2013a), Helen facial (Le et al. 2012b), images of which are taken under uncontrolled conditions. Considering the

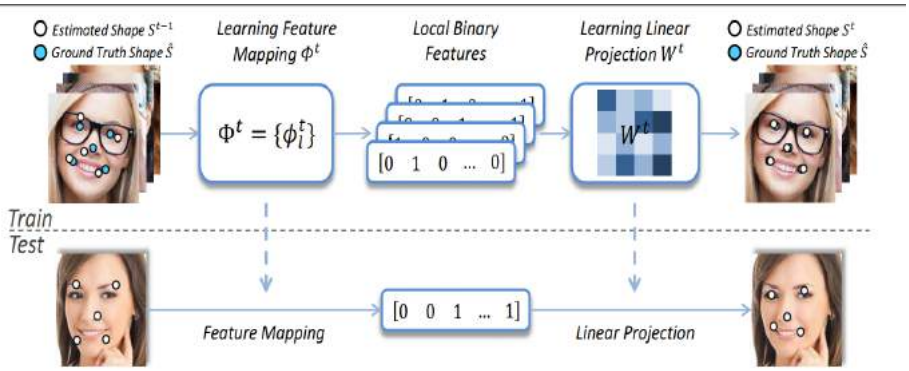


Figure 2.5: *Regression based methods (Ren et al. 2014) learn a feature mapping function $\Phi^t(I_i, S_t - 1)$. Given the features and target shape increments $\Delta\bar{S}_i^t = \bar{S}^i - S_i^{t-1}$, a projection function W^t is learned via regression. In the testing phase, the shape increment is directly predicted and applied to update the current estimated shape.*

limited robustness of the pixel intensity features under occlusion and large shape variation (Cao et al. 2014c), to improve the semantic stability of the shape indexed features, Burgos-Artizzu et al. (2013) proposed to use linear interpolation between two neighbor landmarks to replace the original scheme used in Cao et al. (2014c). They also proposed to train separate regressors for different predefined facial regions to account for partial occlusions. The training method in Cao et al. (2014c) is essentially a greedy method which approximate the function mapping from facial image appearance feature, i.e. shape-indexed feature, to facial shape updates.

In the work of Xiong and De (2013), the supervised descent method was developed, which is based on the natural derivation process of the Newton method. Some experimental results have shown that it is difficult for general regression forests to learn the variations of faces with different head poses due to the large texture appearance variance. Dantone et al. (2012) propose a conditional regression forest scheme that extends the idea of Breiman (2001); Criminisi et al. (2012). Each head pose is conditionally trained with different regression forests, which at runtime are chosen from to run based on the head pose prediction.

To sum up, these methods directly learn a mapping function from training data to facial landmarks, they are efficient and accurate in uncontrolled

environment. Similar to previous group of methods, by exploiting the pose information of faces and training discriminate mapping function for faces with large pose variation their performance could be further improved. The second chapter is dedicated to exploit this fact.

2.1.4 Other methods

Recently with the development of machine learning and pattern recognition techniques, more models have been applied to solve this task. Notably, for the task of facial landmark localization, convolutional neural network (Luo et al. 2012; Zhang et al. 2014a), joint face alignment methods (Zhao et al. 2011) and graphical model (Zhu and Ramanan 2012) have been successfully applied. An example of the convolutional neural network method is shown in Fig. 2.6.

Joint face alignment methods jointly aligns a batch of images undergoing a variety of geometric and appearance variations. assume that the images of same face should lie in the same linear subspace and the person-specific space should be proximate the generic appearance space. Zhao et al. (2011) designed a joint AAM method that an effective and efficient method which simultaneously finds the person-specific appearance space and brings the given images into alignment.

Although images of the same face should lie in the same linear subspace and the person-specific space should be proximate the generic appearance space, the low-rank assumption breaks under several common conditions, such as significant occlusion or shadow, image degradation, and outliers. Smith and Zhang (2012) introduced the same shape model combined with a local appearance model into the joint alignment framework. To distinguish the high quality alignments from the poor ones among all these initial estimations, a discriminative face alignment evaluation metric could be used (Viola and Jones 2004). Validated high quality alignments are utilized to improve the accuracy of poor ones through appearance consistency between the

poor estimate and its selected K neighbouring high quality estimates (Zhao et al. 2012).

Tong et al. (2012) proposed a semi-supervised facial landmark localization approach which utilizes a small number of manually labelled images and minimize the distance between labelled and unlabelled images. Joint optimization utilize multiple images and thus is able to reduce ambiguity and leads to better performance, but it is not suitable for real time or in-the-wild problems.

Graphical model-based methods mainly refer to tree-structure-based methods and Markov random field-based methods. Tree-structure-based methods take each facial feature point as a node and all points as a tree. The locations of facial feature points can be optimally solved by dynamic programming. Zhu and Ramanan (2012) proposed a method that is based on a mixture of trees, each of which corresponds to one head pose view. In the training stage, the tree structure is first estimated via ChowLiu algorithm (Chow and Liu 1968). Then a tree-structured pictorial structure (Felzenszwalb and Huttenlocher 2005) is constructed for each view.

Another work that considers both the local characteristics and global characteristics of facial shapes is the bi-stage component-based facial feature point detection method (Huang et al. 2007). The whole face shape is divided into seven parts. The shape of each part is modeled as a Markov network by taking each point as a node. Belief propagation (Murphy et al. 1999; Felzenszwalb and Huttenlocher 2006) is explored to find the locations of these components. Then, configurations of these components are constrained by the global shape prior described by the Gaussian process latent variable model. Graphical models directly model the relationship between individual landmarks instead of considering them separately and relying on point distribution model, it is robust but requires careful design of the graph representation.

Due to the recent advancement in deep learning, researchers have been employing deep neural nets to improve the feature learning in the facial landmark detection task (Zhang et al. 2014a). Wu et al. (2013) explored deep

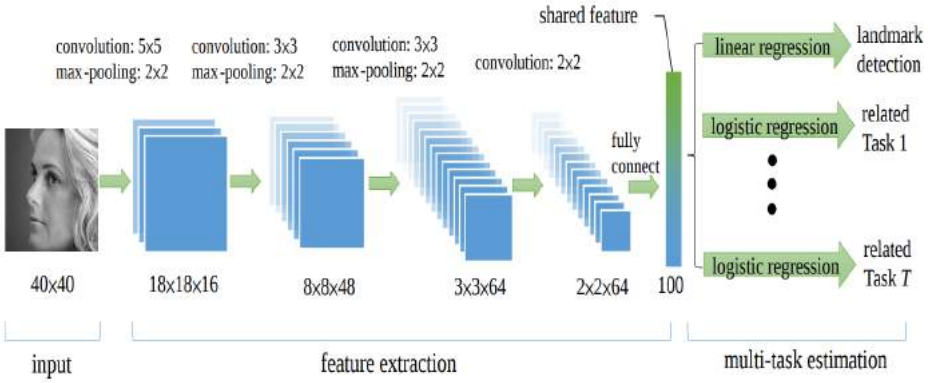


Figure 2.6: *A typical example of deep neural nets trained for the task of facial landmark detection (Zhang et al. 2014a). Robust and discriminative features are learned directly in an end-to-end fashion via gradient descent based methods.*

belief networks to capture face shape variation due to facial expression variations and utilized a 3-way restricted Boltzmann machine (Hinton et al. 2006) to capture the relationship between frontal face shapes and non-frontal face shapes has been applied for facial feature tracking.

Sun et al. (2013) proposed a three-level cascaded deep convolutional network framework for point detection in a coarse-to-fine manner. Each level is composed of several numbers of convolutional networks. The first level gives an initial estimate to the point position and the following two levels then refine the obtained initial estimate to a more accurate one. Though great accuracy can be achieved, this method needs to model each point by a convolutional network which improves the complexity of the whole model. Moreover, with the increase in the number of facial feature points, the time consumption to detect all points is high. Recently, Zhu et al. (2016) proposed a deep learning based 3D solution that employs a strong 3D facial geometry prior to help constrain the problem.

2.1.5 2D Landmark Detection Summary

To summarize, facial landmark detection and localization in images and videos in the wild are a crucial part of reconstructing 3D faces. Currently, existing methods still suffer from large pose variation and partial occlusions. In this

thesis new machine learning techniques to train more discriminative models that deals with these large variations robustly.

2.2 3D Facial Performance Tracking

In order to achieve 3D facial performance tracking, one option is to model the imaging process in order to solve the optimal pose and geometry arrangement that best explain what has been observed in the image. This is a difficult process because the geometry and appearance of human faces vary drastically from person to person and under different environments, which involves a great deal of modelling. Another option is to apply machine learning techniques to learn from a large amount of training samples labelled by animators, which requires a considerable amount of manual labour.

2.2.1 Machine Learning-based Methods

Recently, Cao et al. (2015) has proposed to use recorded wrinkle and image appearance pairs to train a model that is able to capture high frequency details in real time from ordinary cameras, which shows that with a shape prior it is possible to reconstruct detailed structures under uncontrolled setting. In order to achieve realistic results, these methods all require a reference 3D model to regularize and guide the reconstruction process.

Among the state-of-the-art methods, Cao et al. (2013) build a user specific 3D model and collect images processed by 2D landmark and possibly corrected manually to train a 3D shape regressor. Since this method is not fully automatic, in the work of Cao et al. (2014a), the 2D landmarks of thousands of images are manually labeled to recovered their the 3D pose, expression and identity coefficient, then these coefficients and poses are used to train a 3D regressor that can output the expression coefficient and the 3D pose regardless the identity of a particular user. The regressor use shape-indexed pixel difference feature that changes with the pose of user's face, it produce

expression and pose of each frames which are later used to iteratively refine the user's identity coefficient after more frames have been collected.

RGB-based methods (Cao et al. 2014a; Garrido et al. 2013; Weng et al. 2014; Suwajanakorn et al. 2014; Shi et al. 2014) generally rely on a facial landmark detector to coarsely align the face. Numerous techniques have been proposed to locate facial landmarks from image input. These methods reduce the facial alignment error by using a linear combination of the training data or the descent direction to update the average face to explain the input image, e.g., Active Shape Models (Van Ginneken et al. 2002; Cootes et al. 2001), Cascaded Pose Regressors (Dollar et al. 2010; Ren et al. 2014; Cao et al. 2014c), Supervised Descent Methods (Xiong and De 2013; Xiong and De la Torre 2015) and Deformable Part Based Models (Zhu and Ramanan 2012; Tzimiropoulos and Pantic 2014; Ghiasi and Fowlkes 2014). New techniques with higher robustness and accuracy are in constant development. Fundamentally, our method can use any of these facial landmark detection methods for our facial expression transfer.

Thanks to the recently advances in deep learning (Simonyan and Zisserman 2014; Szegedy et al. 2015, 2016), many computer vision researchers have adopted the deep learning approach to tackle the facial tracking problem (Masi et al. 2016; Jackson et al. 2017; Ranjan et al. 2017) with the publicly available large datasets (Zhu et al. 2016; Booth et al. 2016; Shen et al. 2015; Bulat and Tzimiropoulos 2017). These trained deep neural networks have demonstrated impressive results and are very robust in many difficult scenarios compared to traditional machine learning approaches. However, these methods are trained on samples with large variations and try to be robust for every possible scenarios, they usually omit finer details that are important for improving the captured expressiveness facial performances.

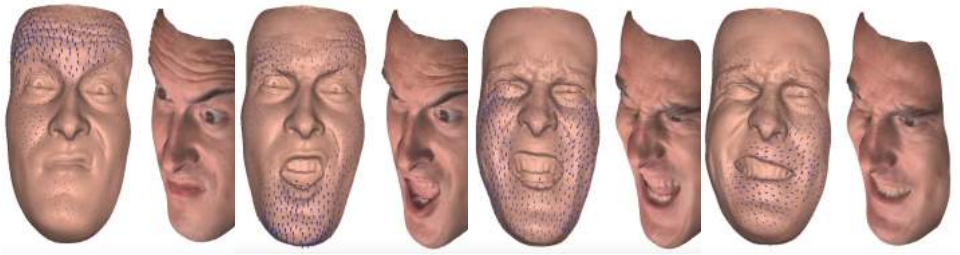


Figure 2.7: *In 3D facial performance tracking, the facial geometry under neutral expression is usually considered as the person-specific canonical reference model. Subsequent facial performance can be tracked and represented as rigid 3D transformation and non-rigid deformation of the reference model. Such deformations from previous frame are shown in the blue vector fields (Gotardo et al. 2015).*

2.2.2 Optimisation-based Methods

In terms of situation where depth information is available, 3D facial mesh tracking becomes a geometry registration problem (Smolyanskiy et al. 2014; Thies et al. 2015). 3D deformation is tracked as shown in Fig. 2.7 to maintain consistent geometry between individual depth map frames. However, a vast majority of videos readily available online or captured by mobile devices are typically not accompanied with depth information. Moreover, since in real world scenarios a person's eye could be occluded by widgets such as transparent glasses, sometimes image based tracking methods have to incorporated together with depth information to yield realistic result.

In terms of tracking with just image color information, once a 3D facial database is available, we can use it as a prior to reconstruct faces from 2D images. Firstly we need to align the reference model to the face in image, which can be formulated as a perspective-n-point (PnP) problem (Gao et al. 2003). Of course due to expression variation further adjustment is needed, but usually the pose of face can be recovered by using the 3D geometry of an average/neutral face.

Other methods such as (Suwajanakorn et al. 2014; Garrido et al. 2013) expands the idea from AAM (Cootes et al. 1999) and use optical flow based method to correct the tracked mesh to be more plausible and expressive. It is achieved via synthesizing images from known model and the minimization of

synthesized result and the real image. Similarly, they need a reference average shape that is be construct before tracking. The illumination parameters can be computed by projecting images of the same person under many different illuminations onto their first four singular vectors (Basri et al. 2007). Dense optical flow (Brox et al. 2004) coupled with shape from shading can recover dense detail and geometry that is lacking in regressor-based 3D facial tracking literatures.

However, these methods are not as robust since they assume a generic lighting model, which leads to distortion in strong specular light, occlusion and shadows. Thies et al. (2016) proposed a dense realtime tracking and facial performance retarding method, which exploits the high computational capacity of moder GPUs. Recently, (Wu et al. 2016) have proposed an anatomically-constrained local deformation model for monocular face capture, which can track facial deformation beyond the predefined blendshapes. Tracking based on anatomically-constrained local deformation model is a break through for monocular RGB methods. Due to the lack of depth information, most of the existing monocular methods constrain the ill-posed problem with blendshapes, which prevents the tracking of deformation not included in the blendshapes.

Various other methods based on computer vision techniques have been introduced. In the work of Valgaerts et al. (2012) for example, a coarse face template is first tracked throughout a binocular stereo sequence, Laplacian deformation model (Sorkine et al. 2004) is used for regularization. Next, shape refinement is performed to recover dense facial details. Weise et al. (2009) investigated structured light approach to reconstruct faces. Typically, for image based methods they all have to trade speed for accuracy and expressiveness due to the lack of depth information.

In addition to the traditional application of 3D facial performance tracking, methods that work with 2D RGB facial geometry also enable a rich number of applications such as digital makeup, face shape editing and facial expression transfer, where examples of such applications are shown in Fig. 2.8.

(a) *Facial expression transfer* **(b)** *Digital makeup*

(c) *Facial feature manipulation* **(d)** *Driving virtual character*

Figure 2.8: *3D facial performance is widely used in many applications such as facial expression transfer (Thies et al. 2016) in Fig. 2.8a, digital makeup (**?**) in Fig. 2.8b, facial component manipulation (Cao et al. 2014b) in Fig. 2.8c and facial animation creation (Cao et al. 2014a) 2.8d. The quality and efficient of 3D facial performance directly play major roles in these applications.*

2.2.3 3D Facial expression transfer

Recently (Cao et al. 2014a) proposed a method that tracks the 3D mesh of a user in real time using RGB cameras. Essentially, the mesh is driven by a linear combination of faces in a 3D face database (Cao et al. 2014b) consisting of different individuals and expressions. However, it mainly focuses on tracking and driving the animation, and not transferring realistic expression onto a real human target from video streams. Thus the applications are limited to driving game characters or virtual avatars.

In order to transfer an expression between the actor and the target, it has to be encoded in the same system. Colour and depth data on their own do not carry much semantic meaning and cannot be used directly to track facial performance. To encode expressions in the same system, virtually all methods are based on the assumption that all faces can be represented by a linear combination of blend shapes or training data. One can adjust the combination to fit the input and derive the expression coefficient from its discrepancy when compared to the neutral expression.

However, unlike depth-based methods, which have access to the depth information, colour-based methods only have access to the projected shape of the face. Generally for these methods, depth is estimated from the same face through different expressions and poses derived from multiple image frames. Since the facial mesh can differ drastically between individuals, some of the colour-based tracking methods require a preprocessing step to build a user specific facial mesh in advance.

Moreover, many existing online image-based facial tracking systems are not able to produce results with high frequency detail, since they only use a coarse set of 2D or 3D facial landmarks. Facial regions that are not tracked by the systems are typically ignored and the information is discarded. Most research on transferring expression to another target focuses on virtual avatars (Kholgade et al. 2011; Weise et al. 2011), where the human actor drives the expression of a virtual character. Creating a realistic morphable model for a real human is difficult due to factors such as skin folding and wrinkling, and change in eye and mouth shape. Transferring expression using only colour information is even more challenging, because without depth information as guidance it has to be robust against depth ambiguity, illumination variation, noisy background and occlusion.

2.2.4 3D Facial Performance Tracking Summary

To summarize, 3D facial performance tracking from RGB input in-the-wild is a difficult task because of the depth ambiguity. Existing methods can be categorized in to optimisation based and machine learning based category. This research will propose a novel optimisation based method to tackle the challenge.

2.3 3D Facial Geometry Reconstruction

Since this work focuses on 3D shape reconstruction, the facial expression tracking and the identity optimization of the actor will be its main focus. The plentiful work on realistic texture, wrinkle and hair reconstruction will be not detailedly discussed. The pioneer work of (Blanz and Vetter 1999) on 3D face animation introduced a face model consists of the shape and texture from a set of 200 faces. A morph model was then constructed and used to 3D face reconstruction from images.

2.3.1 Monocular RGB Reconstruction

Since RGB input are not accompanied by depth information, most methods use a 3D facial database built from depth sensors or photogrammetry systems. For mobile devices equipped with only one RGB camera, reconstructing generic 3D objects from 2D observations can be achieved through structure-from-motion or shape-from-shading methods. We do not discuss photometric stereo in this thesis as it requires special setup and capture process, therefore it is not suitable for uncontrolled environments.

Structure-from-motion methods require moving the camera around to observe the object from multiple perspective. Structure-from-motion(SFM) is a photogrammetry method for 3D structure estimation from 2D images coupled with local motion information. It is a well studied field in computer vision and the general concept is shown in Fig. 2.9. Although hardware

based on infrared projector such as Kinect and Primesense are quite robust and work in indoor settings, their resolution and depth quality is also often limited. Therefore, for certain applications it is more desirable to apply photogrammetry based methods on high resolution 2D images (Beeler et al. 2010; Ichim et al. 2015).

Shape-from-shading methods on the other hand, can work with only a single RGB image. Contrary to most of the other three-dimensional reconstruction problems, for example, stereo and photometric stereo. In the shape-from-shading problem, available data and information are minimal. As a consequence, this inverse problem is intrinsically a difficult one. These methods model the lighting and imaging process in order to recover the underlying 3D structure of a single image (Dovgard and Basri 2004; Patel and Smith 2009; Kemelmacher-Shlizerman and Basri 2011). For example, Zhang et al. (2008) tries to estimate the depth from by assuming a Lambertian reflectance model and harmonic representations of lighting. These methods however, often need images in a controlled light setting, and is thus limited from certain types of application.

Since 3D facial database and 3D facial geometry prior are available, it has been demonstrated that one could directly solve for the 3D shape from a single image (Fried et al. 2016), by applying the prior knowledge to fit the captured RGB information approximately. It has also been demonstrated that once the rough 3D shape of the face has been recover via 3D prior, shape-from-shading methods can be applied to recover details such as wrinkles and skin folds (Suwajanakorn et al. 2014).

An important application of facial animation is to transfer the expression of an actor to the target avatar, which requires the separate parametrization of the identity and expression. The expression parameters need to have the same semantic meaning between the actor and target in order for the transfer to work. Vlasic et al. (2005) introduced a bilinear model consists of 15 person and the same 10 expressions. Yin et al. (2006) introduced a model including both the shape and texture of different individuals. Cao et al. (2014b)

introduced a bilinear similar model compressed by multi mode singular value decomposition (SVD), which contains a wider range of different identities and expressions. However, due to the limited expressiveness of PCA, although these models are more compact and have lower computational complexity, they have limited expressiveness to represent particular users' appearance and thus are robust but not very accurate. Building a person specific model can be useful in certain situation (Cao et al. 2013).

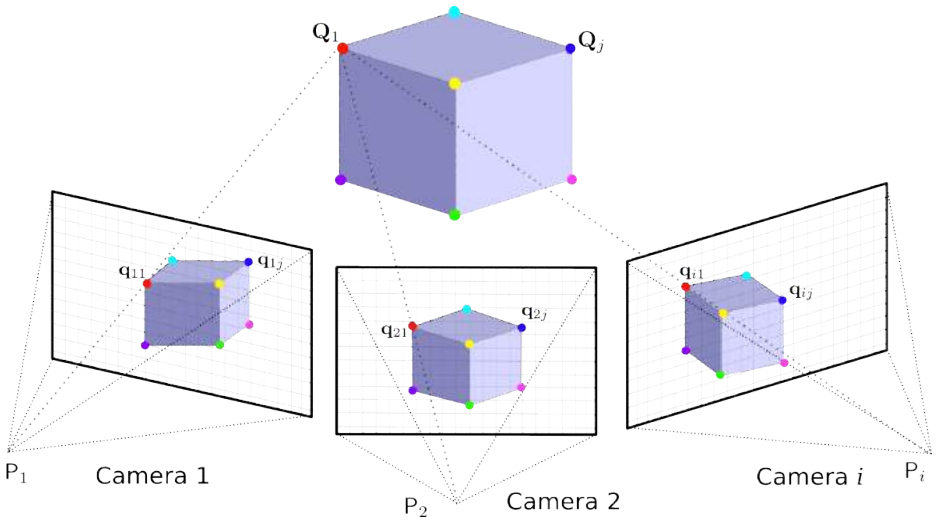


Figure 2.9: *Structure from motion methods estimate the 3D structure of the object captured in 2D images with multiple observations from different viewing perspectives. After establishing correspondences in the images via key points matching, camera position and the 3D information of the key points can be computed via bundle adjustment.*

Ichim et al. (2015) proposed a method that builds dynamic avatars from images captured with hand-held devices, which greatly reduces the cost and hardware requirement for creating highly expressive avatars. SFM based methods estimate the camera position by matching salient points detected in the image. Typically 2D image corner points on edges with strong gradients in multiple directions are usually chosen to be salient points (Lowe 1999). Intuitively corners are more likely to look the same and have the same local patch features under different viewing perspective and distortion, especially in image sources captured by a variety of different cameras (Furukawa and Ponce 2010).

On one hand SFM based methods are very flexible and low cost, intrin-

(a) *Multiview stereo* **(b)** *Photogeometric scene flow*

Figure 2.10: *Multiview stereo setup (Beeler et al. 2010) in Fig. 2.10a and photogeometric setup (Gotardo et al. 2015) in Fig. 2.10b are typically used in professional production to achieve highest possible accuracy. Dedicated hardware and personnel are required to operate these systems.*

sics and extrinsic camera parameters have to be estimated from the key points, which involve solving various time consuming complicated sub-problems and are generally less accurate than systems with calibrated cameras. On the other hand sparse salient points alone are not sufficient for modelling the face where the majority of facial area are smooth varying surface without much texture variation.

2.3.2 Multiview RGB Reconstruction

In controlled studio environments, calibrated multiview camera systems such as shown in Fig. 2.10 are deployed to achieve highest possible accuracy. Laser scanners are generally more expensive and accurate whereas consumer grade products such as Kinect and Primesense are cheaper but noisier. In the work of Newcombe et al. (2011a), a method has been introduced to refine the scans from noisy inputs by sampling repeatedly from the same object. Kinect system is capable of producing real time relative low resolution (640×480) depth maps.

These high quality depth has been used to build 3D facial databases. Template mesh with a predefined set of fixed expressions can be used to guide the generation of expression specific facial mesh generation, together with Kinect fusion which fuse noisy depth maps into a smooth surface. It is usually

achieved by simultaneous deformation and regularization. First the template is deformed to fit the depth map, then by defining correspondence between the facial mesh index and 2D landmarks, one can use deformation algorithms such as the work of Huang et al. (2006) to refine the fitted result. In order to get plausible result, the fitting needs to be regularized by an estimated probability distribution of the shapes.

In order to reconstruct the structure with weak textures, dense depth maps are computed from the solved camera poses via multi-view stereo vision methods such as semi-global-method (Hirschmuller 2008) and patch match (Bleyer et al. 2011; Galliani et al. 2015). These methods exploits the spatial coherence which dictates that neighbouring pixels should have similar depth value unless there are discontinuities indicated by strong edges. In applications where an offline calibration process is possible and ultra fine details are required, for example in (Beeler et al. 2010, 2011; Valgaerts et al. 2012; Gotardo et al. 2015), fully calibrated system are designed with known camera poses and the processing power is directly spent on dense depth map computation. The minimum hardware requirement for these methods is at least two cameras as shown in Fig. 2.11.

Given the pre-recorded 3D facial dataset of many people that cover most of the facial geometry variation between different race, sex and age, it is possible to reconstruct the 3D facial model from several sparse 2D landmarks. However, this limits the detail level of the geometry as unseen variation (glasses, beards, mustaches, nose ring, etc,.) cannot be reconstructed. For these reasons, the geometric details recovered from sparse landmarks are insufficient. Thus in certain works the 3D facial reconstruction process is usually independent from 3D facial performance tracking (Garrido et al. 2013; Wu et al. 2016; Cao et al. 2016).

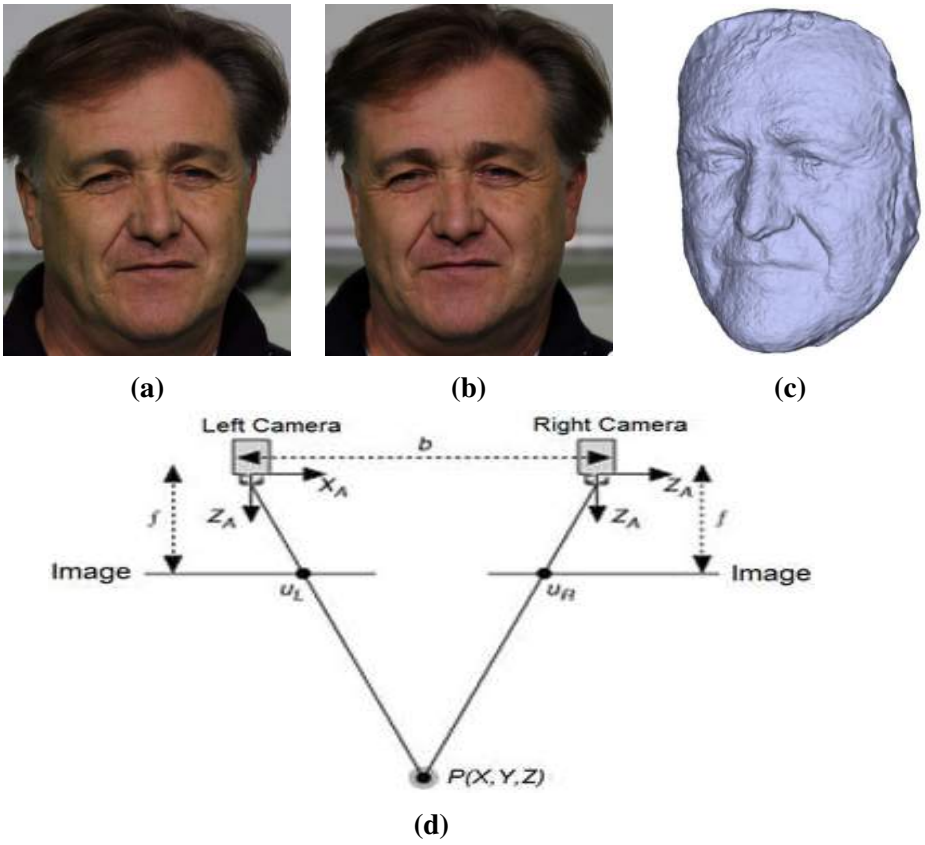


Figure 2.11: *With fully calibrated binocular camera pair with known camera intrinsic and extrinsic parameters (Valgaerts et al. 2012), high quality dense depth map shown in Fig. 2.11c can be directly computed from the left and right view Fig. 2.11a and 2.11b, which is achieved by matching pixels on the same scan-line as shown in Fig. 2.11d.*

2.3.3 3D Facial Geometry Reconstruction Summary

To summarize, 3D reconstruction from 2D input is still an open problem. 3D facial geometry reconstruction as a specific area are challenging due to the fact that people are highly sensitive to even the smallest facial artefacts. This research will propose a novel method to incorporate person specific details into the tracking and reconstruction process to achieve more convincing and realistic results.

Chapter 3

Regression based Facial Landmark Detection for 3D Tracking

Facial landmark detection is widely used for aligning the face template to the real facial pose in the captured image. It aims at locating predefined facial landmarks (such as eye and mouth corners, nose tip) in face images automatically. Face alignment is an important computer vision task, and plays a key role in many applications, such as face recognition, performance-based facial animation, and expression analysis. The alignment usually takes a face bounding box from a face detector as input, and fits initial landmarks positions into optimal locations, which are used to solve for 3D pose of the face, as shown in Fig. 3.1.

Predicting the position of facial landmarks in uncontrolled images is very challenging. In order to tackle this problem, both global or local face appearance features have been considered as constraints for optimization, which are built from sufficient labelled face training images. Recently many methods proposed to solve the face alignment in-the-wild problem, most of them can be categorized into two groups according to the underlying model: generative models and discriminative models. In uncontrolled environments, it's extremely difficult to build generative models that work well because of

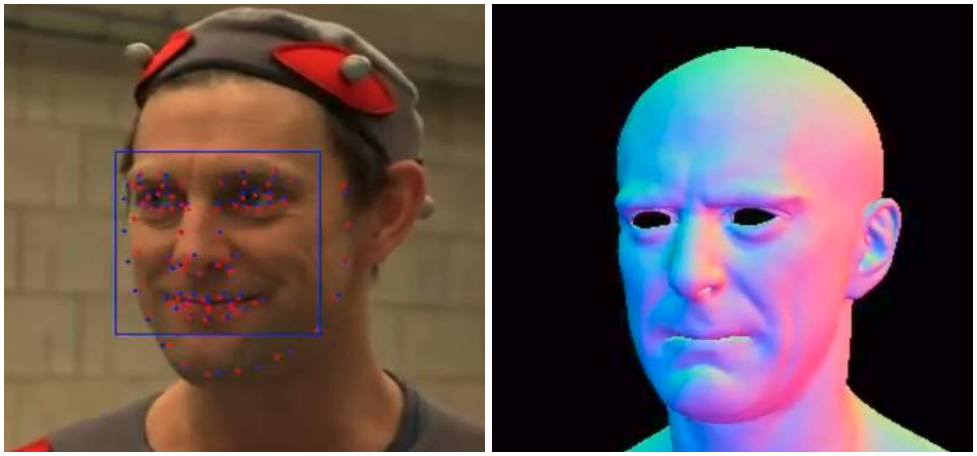


Figure 3.1: *First facial bounding box is detected as shown in the blue rectangle, next the mean 2D facial landmarks computed during training are aligned to the bounding box and updated to match the real pose observed in the image, which are shown with blue points. Finally, 3D pose can be solved by aligning the 3D model to the 2D points, whose 2D projections are shown in red points.*

the large variation of appearance, lighting and various other factors. Discriminative models solve the problem by learning from a large amount of training samples and have been proven to work for in uncontrolled environments. However, most existing methods model the facial landmark as 2D shape, hence they cannot handle large expression variations under large pose variations. To this end, we propose two novel landmark detection methods that model the landmarks with consideration of their inherent 3D structure.

The structure of this chapter is organised as follows: In Section 3.1 we introduce our novel sign-correlation partition scheme, which improves the facial landmark detection performance when employed in the global supervised descent method framework by improving the training sample partition results with large pose variations (Xiong and De la Torre 2015). In Section 3.2 we introduce a bilinear 3D model that directly learns the facial landmarks with a 3D representation.

3.1 Sign-correlation partition for global supervised descent method

Even though previous works have produced remarkable results on nearly frontal face alignment, it is still hard to locate landmarks across large poses and expressions under uncontrolled conditions. The variation of poses leads to non-convex and multiple local minimum problems. Especially, in the work of Xiong and De la Torre (2015), the problem was theoretically addressed and a descent domain partition was proposed that models the features extracted from the training images and the facial landmark shape PCA space separately. Though their scheme works well for face tracking and pose estimation, it is not suitable for face alignment across various poses. Because the ground truth of shapes or features are unknown at teat time, this scheme fails to find a proper approximation due to the lack of initialization from previous frames.

A few recent works (Burgosartizzu et al. 2013; Xing et al. 2014; Feng et al. 2015b; Yang et al. 2015b; Zhu et al. 2015; Feng et al. 2015a) begin to consider the influence of multiple poses. Most of them deal with the problem indirectly by data augmentation which samples many poses randomly, but they can only handle small changes in poses. How to solve non-convex and multiple local minimum problems caused by large poses is still not well studied. Although the state-of-the-art Supervised Descent Method (SDM) has shown decent results for frontal poses, it cannot handle large pose and expression variation because it learns conflict descent maps in the whole complex space. Global SDM has been presented to deal with this case by domain partition in feature and shape PCA spaces for face tracking and pose estimation. However, it is not suitable for the face alignment problem due to unknown ground truth shapes. Some current researches like random subspace SDM, Stage-wise Relational Dictionary, and coarse-to-fine shape searching can mitigate multi-pose face alignment problem, but there is still no a method that deal with the multiple local minima problem directly.

Therefore in this section we propose a sign-correlation subspace method

for domain partition in only one reduced low dimensional subspace, which can handle the multiple local minima problem. Inspired by (Xiong and De la Torre 2015), we proposed a novel sign-correlation subspace method for partitioning descent domains to achieve robust face alignment across poses. Unlike previous methods, we analyse the sign correlation between features and shapes, and project both of them into a mutual sign-correlation subspace. Each pair of projected shape and feature keep sign consistent in each dimension of the subspace, so that each hyperoctant holds the condition that one general descent exists. Then a set of general descents are learned from the samples in different hyperoctants. Since only the feature projection is needed for domain partition, our method is better suited for discovering good features for face alignment across large pose variations. Our sign-correlation partition method is evaluated on public face datasets, which includes a range of poses. The experimental results indicate that our methods can reveal their latent relationships to poses, as shown in Fig. 3.2 and 3.3. The comparison with state-of-the-art methods for face alignment demonstrates that our method outperforms them especially in uncontrolled conditions with various poses, while keeping comparable speed.

The main contributions of the proposed methods are:

- The inherent relationship between poses space and appearance features or shapes space is explicitly obtained by sign-correlation reduced dimension strategy. The whole features and shapes spaces are projected into a mutual sign-correlation subspace, which mainly represents the variation of poses.
- The decent domains partition is produced according to the signs of each dimension in this sign-correlation subspace. The decent domains partition is computed via projecting features space into joint sign-correlation subspace and splitting whole sample space into different hyperoctants as decent domains, which provides better partition than naive PCA partitionn.
- Our method is evaluated on public face benchmark datasets, which

includes face images from different poses. The results show that it can split complex sample space into homogeneous domains related to poses, thus a mutual manifold of feature and shape spaces is obtained.

The experiments for face alignment indicate that our method get state-of-the-art performance for nearly frontal face images, and it is more accurate on datasets with multiple poses.

3.1.1 Sign-correlation Subspace for descent domains partition

Descent domain partition problem in SDM

In order to keep the completeness of our work, we first review the problem of SDM and the sign-correlation condition for existence of a supervised descent domain. According to SDM, given one image **d**, p landmarks $\vec{x} = [x_1, y_1, \ldots, x_p, y_p]$, a feature mapping function $\vec{h}(\mathbf{d}(\vec{x}))$, where $\mathbf{d}(\vec{x})$ indexes landmarks in the image **d**, thus face alignment can be treated as an optimization problem (Xiong and De la Torre 2015),

$$f(\vec{x}_0 + \Delta\vec{x}) = \|\vec{h}(\mathbf{d}(\vec{x}_0 + \Delta\vec{x})) - \phi_*\|_2^2 \tag{3.1}$$

Where $\phi_* = \vec{h}(\mathbf{d}(\vec{x}_*))$ represents the feature extracted according to correct landmarks $\vec{x}_*$, which is known in the training images, but unknown in the testing images. For an initial locations of landmarks $\vec{x}_0$, we solve $\Delta\vec{x}$, which minimizes the feature alignment error $f(\vec{x}_0 + \Delta\vec{x})$. Since the feature function is usually not analytically differentiable, it is hard to solve the problem with traditional Newton methods. Alternatively, a general descent mapping can be learned from training datasets. The supervised descent method form for regression level k is

$$\vec{x}_k = \vec{x}_{k-1} - \mathbf{R}_{k-1}(\phi_{k-1} - \phi_*) \tag{3.2}$$

Since ϕ_* of a testing image is unknown but constant, SDM modifies the objective to align with respect to the average one $\overline{\phi}_*$ over training set, the update rule then is modified,

$$\Delta\vec{x} = \mathbf{R}_k(\overline{\phi_*} - \phi_k) \tag{3.3}$$

Instead of learning only one $\mathbf{R}_k$ over all samples during one updating step, The Global SDM learns a series of $\mathbf{R}^t$, one for a subset of samples S^t, where the whole dataset is divided into T subsets $S = \{S^t\}_1^T$.

A generic descent direction exists under these two conditions: 1) $\mathbf{R}\vec{h}(\vec{x})$ is a strictly locally monotone operator anchored at the optimal solution 2) $\vec{h}(\vec{x})$ is locally Lipschitz continuous anchored at $\vec{x}_*$. For a function with only one minimum, these normally hold. But a complex function might have several local minimum in a relatively small neighborhood, thus the original SDM tends to average conflicting gradient directions. Therefore, the Global SDM proves that if the samples are properly partitioned into a series of subsets, there is a descent direction in each of the subsets. The $\mathbf{R}_t$ for subset S_t can be solved with a constrained optimization form,

$$\min_{S,\mathbf{R}} \sum_{t=1}^{T} \sum_{i \in S^t} \|\Delta\vec{x}_*^i - \mathbf{R}_t \Delta\phi^{i,t}\|^2 \tag{3.4}$$

$$s.t.\ \Delta\vec{x}_*^i \mathbf{R}_t \Delta\phi^{i,t} > 0, \forall\, t, i \in S^t \tag{3.5}$$

Where $\Delta\vec{x}_*^i = \vec{x}_*^i - \vec{x}_k^i$, $\Delta\phi^{i,t} = \overline{\phi}_*^t - \phi^i$, where $\overline{\phi}_*^t-$ average all ϕ_* over the subset S^t. Eq.3.5 guarantees that the solution satisfies descent direction condition 1. It is NP-hard to solve Eq.3.4, so a deterministic scheme is proposed to approximate the solution. A set of sufficient conditions for Eq.3.5 is given:

$$\Delta\vec{x}_*^{i^T} \Delta\mathbf{X}_*^t > \vec{0}, \forall\, t, i \in S^t \tag{3.6}$$

$$\Delta\Phi^{t^T} \Delta\phi^{i,t} > \vec{0}, \forall\, t, i \in S^t \tag{3.7}$$

Where $\Delta\mathbf{X}_*^t = [\Delta\vec{x}_*^{1,t}, \ldots, \Delta\vec{x}_*^{i,t}, \ldots]$, each column is $\Delta\vec{x}_*^{i,t}$ from the subset

S^t; $\Delta\Phi^t = [\Delta\phi^{1,t}, \ldots, \Delta\phi^{i,t}, \ldots]$, each column is $\Delta\phi^{i,t}$ from the subset S^t.

Since the dot product of any two vectors within the same hyperoctant (the generalization of quadrant) is positive, a ideal sufficient partition can be like that each subset S^t occupies a hyperoctant both in the parameter space $\Delta\vec{x}$ and feature space $\Delta\phi$. However, this leads to exponential number of descent directions. Assuming $\Delta\vec{x}$ is n-dimension, and $\Delta\phi$ is m-dimension, the number of subsets will be 2^{n+m}. Moreover, if the number of all samples is small, there will be many empty subsets, and also the volume of some subsets will be too small to train.

It is known that $\Delta\vec{x}$ and $\Delta\phi$ are embedded in a lower dimensional manifold for human faces. So dimension reduction methods(e.g. PCA) on the whole training set $\Delta\vec{x}$ and $\Delta\phi$ can be used for approximation. The Global SDM authors project $\Delta\vec{x}$ onto the subspace expended by the first two components of $\Delta\vec{x}$ space, and project $\Delta\phi$ onto the subspace by the first component of $\Delta\phi$ space. So there are 2^{2+1} subsets in their work. As shown in the experimental results, this scheme performs poorly on faces with large pose variations. In order to solve this, we introduce correlation-based dimension reduction theory to develop a more practical and efficient strategy for low-dimension approximation of the high dimensional partition problem.

Sign-correlation Subspace Partition

In the work of Xiong and De la Torre (2015), it has been proved that if one subset S^t satisfies: For any two samples $\{\Delta\vec{x}^{i,t}, \Delta\phi^{i,t}\}$, $\{\Delta\vec{x}^{k,t}, \Delta\phi^{k,t}\}$ within S^t, the signs of each corresponding $j-th$ dimension $\{\Delta x_j^{i,t}, \Delta\phi_j^{i,t}\}$ between the samples keep the same,

$$sign(\Delta x_j^{i,t}, \Delta\phi_j^{i,t}) = sign(\Delta x_j^{k,t}, \Delta\phi_j^{k,t}), \forall i, k \in S^t, j = 1 : min(n, m) \tag{3.8}$$

Then there must exist a descent direction $\mathbf{R}^t$ in one updating step. Eq.3.8 provides a possible partition strategy: all the samples that follow Eq.3.8 can be put into a subset, and there would be $2^{min(n,m)}$ subsets in total. It is noticed

that there are two limitations of this partition strategy: 1) It can not guarantee the samples lie in the same small neighbourhood. In other words, even if $\{\Delta\vec{x}^{i,t}, \Delta\phi^{i,t}\}$, $\{\Delta\vec{x}^{k,t}, \Delta\phi^{k,t}\}$ keep Eq.3.8, the $\Delta\vec{x}^{i,t}$, $\Delta\vec{x}^{k,t}$ may be very far from each other. 2) It only considers the dim-to-dim correlation of the first $\min(n, m)$ dimensions in the $\Delta\vec{x}$ space and $\Delta\phi$ space, and other dimensions are ignored. The correlation of any $j-th$ dimension of $\Delta\vec{x}$ with a non-corresponding $j^{'}-th$ dimension $j^{'} \neq j$ of $\Delta\phi$ is also ignored.

To this end, we take the low dimensional manifold into consideration, and show that the $\Delta\vec{x}$ space and $\Delta\phi$ space can be projected onto a medium low dimensional space with projection matrix $\mathbf{Q}$ and $\mathbf{P}$, respectively, which keeps the projected vectors $\vec{v} = \mathbf{Q}\Delta\vec{x}$, $\vec{u} = \mathbf{P}\Delta\phi$ correlated enough: 1) $\vec{v}$, $\vec{u}$ lie in the same low dimensional space. 2) For each $j-th$ dimension, $sign(v_j, u_j) = 1$. If the projection holds these two conditions, the projected samples $\{\vec{u}^i, \vec{v}^i\}$ can be partitioned into different hyperoctants in the medium space only according to the signs of $\vec{u}^i$, due to condition 2. Since samples in a hyperoctant are close enough to each other, this partition can hold the small neighbourhood better. It is also a compact low dimensional approximation of the high dimensional hyperoctant-based partition strategy in both $\Delta\vec{x}$ space and $\Delta\phi$ space, which is an sufficient condition for the existence of a generic descent direction, as mentioned above.

For convenience, we denote $\Delta\vec{x}$ as $\vec{y} \in \Re^n$, $\Delta\phi$ as $\vec{x} \in \Re^m$, $\mathbf{Y}_{s\times n} = [\vec{y}^1, \ldots, \vec{y}^i, \ldots, \vec{y}^s]$ is all the $\vec{y}^i$ of training set. $\mathbf{X}_{s\times m} = [\vec{x}^1, \ldots, \vec{x}^i, \ldots, \vec{x}^s]$ is all the $\vec{x}^i$ of training set. The projection matrices are

$$\mathbf{Q}_{r\times n} = [\vec{q}_1, \ldots, \vec{q}_j, \ldots, \vec{q}_r]^T,\ \vec{q}_j \in \Re^n,$$

$$\mathbf{P}_{r\times m} = [\vec{p}_1, \ldots, \vec{p}_j, \ldots, \vec{p}_r]^T,\ \vec{p}_j \in \Re^m,$$

Projection vectors are $\vec{v} = \mathbf{Q}\vec{y}$, $\vec{u} = \mathbf{P}\vec{x}$. Here we denote projection vectors $\vec{w}_j$, $\vec{z}_j$ along the sample space: $\vec{w}_j = \mathbf{Y}\vec{q}_j = [v_j^1, \ldots, v_j^i, \ldots, v_j^s]^T$, $\vec{z}_j = \mathbf{X}\vec{p}_j = [u_j^1, \ldots, u_j^i, \ldots, u_j^s]^T$. This problem can be formulated as a

constrained optimization form,

$$\min_{\mathbf{P},\mathbf{Q}} \sum_{j=1}^{r} \|\mathbf{Y}\vec{q}_j - \mathbf{X}\vec{p}_j\|^2 = \min_{\mathbf{P},\mathbf{Q}} \sum_{j=1}^{r}\sum_{i=1}^{s} (v_j^i - u_j^i)^2 \tag{3.9}$$

$$s.t. \sum_{j=1}^{r}\sum_{i=1}^{s} sign(v_j^i u_j^i) = sr \tag{3.10}$$

It can be seen that $\vec{w}_j$ and $\vec{z}_j$ are the projected values of all the samples $\mathbf{Y}$ or $\mathbf{X}$ along a special direction $\vec{q}_j$ or $\vec{p}_j$. For a fixed projected $j-th$ dimension, assuming the $\vec{w}_j$ and $\vec{z}_j$ is normalized, which means that the mean of $\{v_j^i\}_{i=1:s}$ is zero, and the standard deviation of it is $1/s$, so $\{u_j^i\}_{i=1:s}$ is. Thus $\vec{w}_j^T\vec{w}_j = 1$, $\vec{z}_j^T\vec{z}_j = 1$, $\vec{w}_j^T\vec{e} = 0$, $\vec{z}_j^T\vec{e} = 0$, where $\vec{e} = [1, 1, \ldots, 1]^T$, then Equation. 3.9 can be simplified as,

$$\min_{\mathbf{P},\mathbf{Q}} \sum_{j=1}^{r} \|\vec{w}_j - \vec{z}_j\|^2 = \max_{\mathbf{P},\mathbf{Q}} \sum_{j=1}^{r} \vec{w}_j^T\vec{z}_j \tag{3.11}$$

For a fixed projected $j-th$ dimension, the constraint $\sum_{i=1}^{s} sign(v_j^i u_j^i) = s$ means that all the pairs $\{v_j^i, u_j^i\}$ of samples in $j-th$ dimension keeps the consistence of sign.There is a fact: if the angle θ_j between $\vec{w}_j$ and $\vec{z}_j$ is 0, the term $\vec{w}_j^T\vec{(}z_j$ will reach maximum, so the sign condition must hold; and if the angle θ_j is $\pi/2$, the Eq.3.12 will reach 0, so the sign condition will fail completely. Moreover, fixing the $|v_j^i u_j^i|$, the $\cos\theta_j$ will get larger while the $\sum_{i=1}^{s} sign(v_j^i u_j^i)$ rises, and $\sum_{i=1}^{s} sign(v_j^i u_j^i)$ tends to larger with the $\cos\theta_j$ growing. Given some constraints, it can be proved that the $\cos\theta_j$ can be taken as an approximation of the sign summation function for optimization,

$$\frac{1}{s}\sum_{i=1}^{s} sign(v_j^i u_j^i) \approx \cos\theta_j = \vec{w}_j^T\vec{z}_j \tag{3.12}$$

When the samples $\{\vec{y}^i\}_{i=1:s}$ and $\{\vec{x}^i\}_{i=1:s}$ are normalized(removing means and dividing standard deviation during preprocessing), the sign-correlation constrained optimization problem will be solved with the standard Canonical-Correlation Analysis(CCA). The CCA problem for normalized $\{\vec{y}^i\}_{i=1:s}$ and

$\{\vec{x}^i\}_{i=1:s}$ is,

$$\max_{\vec{(p)}_j,\vec{q}_j} \vec{q}_j^T cov(\mathbf{Y},\mathbf{X})\vec{p}_j \tag{3.13}$$

$$s.t.\ \vec{q}_j^T var(\mathbf{Y},\mathbf{Y})\vec{q}_j = 1,\ \vec{p}_j^T var(\mathbf{X},\mathbf{X})\vec{p}_j = 1 \tag{3.14}$$

Following CCA algorithm, the maximum sign-correlation dimension $\vec{p}_1$ and $\vec{q}_1$ is solved at first. Then one seeks $\vec{p}_2$ and $\vec{q}_2$ by maximizing the same correlation subject to the constraint that they are to be uncorrelated with the first pair $\vec{w}_1$, $\vec{z}_1$ of canonical variables; This procedure may be continued up to r times until $\vec{p}_r$ and $\vec{q}_r$ is solved.

After all $\vec{p}_j$ and $\vec{q}_j$ is solved, we only need the projection matrix $\mathbf{P}$ in $\Delta\vec{x}$ space. Then we project each $\Delta\vec{x}^i$ into the sign-correlation subspace and get reduced feature $\vec{u}^i = \mathbf{P}\Delta\vec{x}^i$. Then we partition the whole sample space into independent descent domains by judging the sign of each dimension of $\vec{u}^i$ and group it into corresponding hyperoctant. Finally, in order to solve Eq.3.4 at each iterative step, we learn a descent mapping for every subset at each iterative step with the ridge regression algorithm. When testing a face image, we also use the projection matrix $\mathbf{P}$ to find its corresponding decent domain and predict its shape increment at each iterative step.

3.1.2 Experiments and Evaluation

Our work mainly focuses on face alignment across poses, so we conduct experiments especially for this task to analyse and evaluate our sign-correlation partition method. Firstly, we evaluated our method on multi-pose datasets by comparing our method to the PCA partition scheme. Then we also test our method on common datasets for general face alignment and compare it with state-of-the-art methods. According to the work of Yan et al. (2013a), the multi-scale HOG outperform multi-scale SIFT and other typical local descriptors HOG, SIFT, LBP and Gabor. We adopt multi-scale HOG as feature mapping function in our sign-correlation partition SDM algorithm. The two domains are enough for partition, so we only use the first sign-correlation

projection component in appearance feature space.

Sign-correlation partition

In this section, we validate the underlying relationship between our sign-correlation partition and the variation of poses. Two widely used benchmark datasets are used in our validation: MTFL (Zhang et al. 2014b) and 300W (Sagonas et al. 2013b). MTFL dataset contains labeled face images from AFLW (Kostinger et al. 2011), LFW (Huang et al. 2008) and Internet. This dataset annotates 5 landmarks and labels 5 different left-right poses with the flags of gender, smile and glasses.

Here we only focus on non-frontal poses to verify our partition method. There are 2550 non-frontal face images in original MTFL dataset, and the number of left ones is not equal to the number of right ones. For fairness, we augment these non-frontal image by a horizontal flip, so that we get the same numbers of left and right images. 300W dataset is mainly made up of images from LFPW (Belhumeur et al. 2013), HELEN (Le et al. 2012b), AFW(Ramanan 2015) with 68 re-annotated landmarks. The 3148 images from training dataset are selected in our validation. The flip augment is also used for obtaining the same number of left and right images. The left or right poses are estimated by a typical pose estimation taking known landmarks as input.

We partition the multi-poses images into two domains by the first sign-correlation projected dimension. The PCA partition by the first principle component is also tested as comparison. The results in Fig. 3.2 and Fig. 3.3 show that each sign-correlation domain mainly contain left or right pose images, and the accuracy of pose partition is high, as shown in Table 3.1. It indicates that our sign-correlation partition method can construct descent domains highly related to pose variations only with face appearance features. On the contrary, PCA partition only with face appearance features can not capture the pose variation well, and the partition result is nearly random.

Figure 3.2: *Pose validation on MTFL. The average faces are shown in the leftmost column, and examples faces corresponding the average face are shown in the following columns in the same row. The top 2 rows shows the partition results of our method, and the bottom 2 rows shows the results from PCA partition.*

Comparison of face alignment

The proposed sign-correlation partition SDM method was evaluated on the challenging 300W dataset. There are 68 labeled landmarks in this dataset. Its training part contains 3148 images form AFW and training parts of LFPW, and HELEN dataset, and its testing part contains 689 images from testing parts of LFPW, and HELEN and IBUG. Among them, the LFPW dataset, although more challenging than other near-frontal datasets, is mainly made up of small pose variations, and the result on it nearly reaches limitation. The HELEN dataset contains faces of different genders, poses, and expressions. The IBUG testing dataset is most challenging one due to extreme poses, ex-

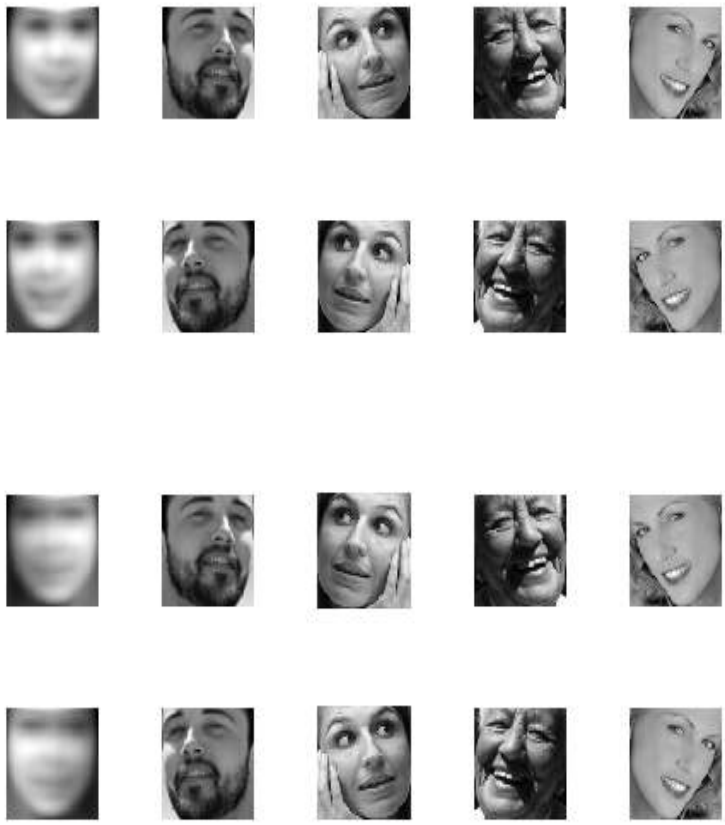


Figure 3.3: *Pose validation on 300W. The average faces are shown in the left-most column, and examples faces corresponding the average face are shown in the following columns in the same row. The top 2 rows shows the partition results of our method, and the bottom 2 rows shows the results from PCA partition.*

pressions and lighting. Proposed method was compared to state-of-the-art methods ESR (Cao et al. 2014c), SDM (Xiong and De 2013), ERT (Kazemi and Sullivan 2014), and LBF (Ren et al. 2014).

We conduct three experiments by testing different parts of the whole 300W datset: common subset: LFPW and HELEN, challenging dataset: IBUG, and full dataset. Following the standard (Huang et al. 2008), the normalized inner-pupil distance landmark error is used in our evaluation. The inner-pupil errors of different methods are given in Table 3.2. The cumulative error distribution (CED) curves are also plotted, as shown in Fig. 3.4. The results illustrate that our method outperform most of current methods over the

Datasets	PCA	Sign-Correlation
MTFL	0.7780	**0.9275**
300W	0.5179	**0.9319**
HELEN	0.7133	**0.9635**
LFW	0.4125	**0.9511**

Table 3.1: *The pose classification accuracy of PCA and Sign-Correlation is validated on MTFL, 300W, HELEN and LFW. The statistics clearly show that the proposed Sign-Correlation method outperforms vanilla PCA in the pose classification task.*

Table 3.2: *Comparison with current methods on 300W dataset*

Datasets	Full	Common	Challenging
ESR	7.58	5.28	17.00
SDM	7.52	5.60	15.40
ERT	6.41	5.22	13.03
LBF	6.32	4.95	11.98
Ours	**5.88**	5.07	**10.79**

full datasets. We also get comparable results on common LFPW and HELEN datasets. Our method works better especially on the challenging IBUG dataset with large variations of poses.

3.1.3 Summary

We propose a novel sign-correlation partition method for global SDM algorithm, and achieve promising results for face alignment on the benchmark datasets. We analyze the underlying relationship between shape/feature space and pose space by sign-correlation reduced dimensional projection. Taking the advantage of the inherent connection of shapes with features within a mutual pose-related subspace, the global descents partition can be operated according to different hyperoctants in the projected sign-correlation subspace. Due to the high consistence of sign between shapes and features in this subspace, it is able to partition the descent domains only depends on features and learned sign-correlation projection components.

Moreover, we provide a clearer explanation for the influence of poses on the problems of multiple local minimum and global descent mapping con-

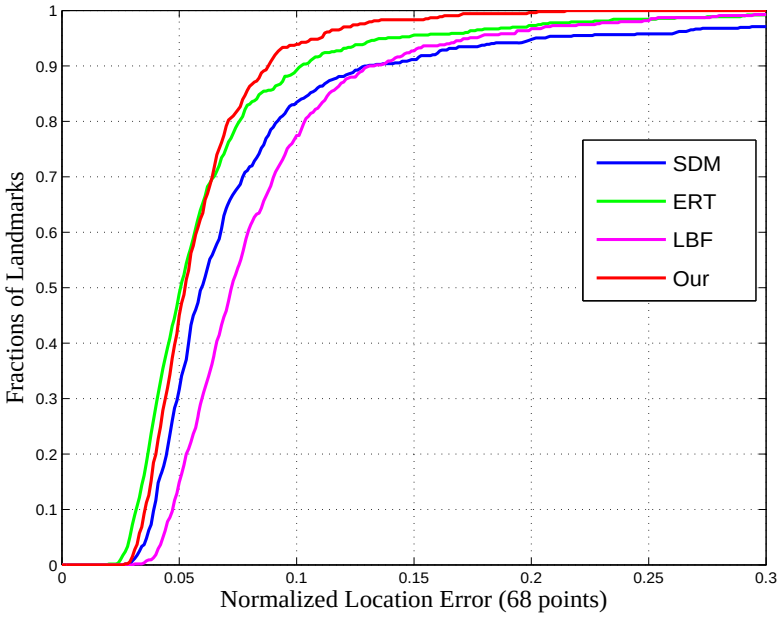


Figure 3.4: *The CED curve of SDM, ERT, LBF and our method on the 68 points 300W markup scheme, which shows that our method outperforms the state-of-the-art methods on this task.*

flict and tackle with the robust face alignment across poses in a direct way. The experiment on the widely used multi-pose dataset indicates that our sign-correlation partition method can divide global complex space into several pose-related descent domains only with appearance features rather than PCA partition in both shape and feature spaces. Our method also achieves noticeable results for face alignment on benchmark datasets compared to popular methods.

Although our sign-correlation subspace method improves the performance in extreme conditions, there are still some parts we need to study in the future: The number of sign-correlation dimensions can be chosen in a more mathematically sound way to ensure optimum performance. Since the partition accuracy is limited by linear reduced dimensional projection, the kernel method can also be introduced into sign-correlation analysis.

3.2 3D bilinear model for supervised descent method

2D based methods are inherently weaker than 3D based methods on 3D object landmark detection such as facial landmark detection. Existing data-driven methods for monocular videos suffer from large variations of pose and expression. To tackle these challenges, we propose a more efficient strategy for solving this task by introducing a novel supervised coordinate descent method with 3D bilinear representation.

Instead of learning the mapping between the whole parameters and image features directly with a cascaded regression framework in existing methods (Xiong and De 2013; Xiong and De la Torre 2015), we learn individual sets of parameters mappings separately step by step by a coordinate descent mean. Because different parameters make different contributions to the displacement of facial landmarks, our method is more discriminative to current cascaded regression methods that consider different sets of parameters as a whole.

Benefiting from a 3D bilinear model learned from public databases, the proposed method can handle the head pose changes and extreme expressions out of plane better than other 2D-based methods. Our method delivers reliable face tracking under various head poses and facial expressions on uncontrolled video sequences collected online. The experimental results show that our method outperforms state-of-art data-driven methods.

In the past, some model-based methods have been studied for face alignment in 2D image domain. One well-known method is active shape model (ASM)(Cootes et al. 1999), which is one of the earliest data-driven models for shape fitting. As an improvement, active appearance model (AAM) (Cootes et al. 2001) considers the global appearance rather than only local textures in ASM. Although AAM and its variations (Cootes et al. 2002; Cristinacce and Cootes 2006; Gonzalezmora et al. 2007; Lee and Kim 2009) can fit face well for near-frontal face images, they tend to fail for uncontrolled face images in the wild environment.

Recently, regression-based methods have been exploited, and cascaded regression with shape-indexed feature is introduced for face alignment, like face alignment by explicit shape regression (Cao et al. 2014c) and supervised descent method (Xiong and De 2013). These methods are efficient enough for real-time face fitting, but they cannot capture variations of pose and expression out of plane due to lack of 3D information.

With the development of commercial depth acquisition devices, it has been convenient to obtain 3D face data. Benefiting from existing 3D face databases, regression-based methods with 3D information have been proposed. While some methods(Feng et al. 2015a) uses 3D face database for augmenting 2D face images with wilder ranges of poses and expressions, some methods have trained 3D face models for user-specific multi-parameter regression methods (Cao et al. 2014c). At present, this kind of method can deal with slight variations of pose and expression in 3D space, while a post-processing is still needed.

In this section, we are aiming for low cost commodity webcams that make accurate face alignment and tracking much harder. We propose a novel supervised coordinate descent method combined with a 3D bilinear face model for robust face alignment and tracking across poses and expressions. The main contribution is a novel supervised coordinate descent method for learning different types of parameters mappings separately to reduce the cross impact caused by learning a whole-parameter mapping. Taking advantage of a 3D bilinear model learned from public databases, our method can handle large pose and expression variations in 3D space with high accuracy. Our method is evaluated in benchmark face videos collected from the Internet.

3.2.1 Supervised coordinate descent method with a 3D bilinear model

Our work is inspired by 3D Dynamic Displacement Model and 2D-based cascaded regression. Since different parameters have different impact on 2D

displacements of landmarks, training a cascaded regression directly for the whole parameters probably leads to cross impact, thus there is still a complex post-processing needed for refine the coarse regressed result. In this section a supervised coordinate decent method is proposed to learn the different types of parameters.

We present a 3D multi-parameters cascaded regression method based on a 3D Bilinear Model and Dynamic Displacement Model. The 3D Bilinear Model describes a bilinear representation of a 3D face mesh, and the Dynamic Displacement Model describes how to project 3D landmarks onto their corresponding 2D ones on the image plane. The parameters include pose matrix of perspective projection, expression coefficients and identity coefficients based on a trained 3D bilinear model, and 2D displacements based on Dynamic Displacement Model. The key idea in this work is that different kinds of parameters mapping with shape-indexed should be learned one by one through a supervised coordinate descent way. It means that other parameters keep fixed while learning a specific parameter mapping at a regression step. A flowchart of our supervised coordinate descent method is shown in Fig. 3.5.

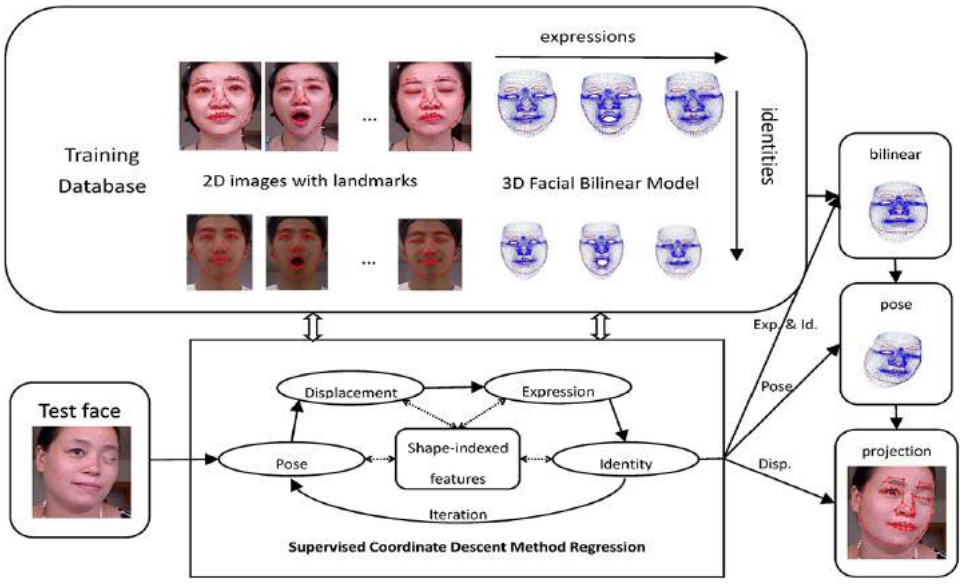


Figure 3.5: *Overview of supervised coordinate descent method. Given the training images and 2D facial landmarks ground truth annotated manually, we fit a 3D bilinear facial model to the 2D landmarks and learns the 3D rotation, translation, expression and identity parameters separately. At test time the learned model extracts the features from the image and predicts the 3D pose parameters.*

3D bilinear model and 2D displacement model

Based on the work of Vlasic et al. (2005) and Cao et al. (2014a), a 3D face mesh can be represented as a bilinear combination by identity expression modes. When the expression keeps fixed, a use-specific face can be represented as a linear combination of faces with different identities; when the identity keeps fixed, a face with specific expression can be represented as a linear combination of faces with a series of predefined expressions. A 3D face database with different identities and expressions can be represented as a three modes tensor by vertex $\times$ identity $\times$ expression. And N-mode SVD (De Lathauwer et al. 2000) is used to compress the huge tensor to a small core tensor. For a 3D face V with a dimension of 11510×3, its bilinear representation is described as Eq. 3.15 (Cao et al. 2014b):

$$V = C_r \times_2 u^T \times_3 e^T \tag{3.15}$$

where C_r is a core tensor with a dimension of $11510 \times 50 \times 25$, u is the 50×150 identity matrix, e is the 25×47 expression matrix, and $\times_k$ is product operator by mode-k.

The transformation from object coordinate to camera coordinate is obtained by a pose matrix including 3D rotation and translation is shown as follows:

$$F = \mathbf{R}V + \mathbf{t}, \tag{3.16}$$

where F is a 3D face mesh in camera coordinate, $\mathbf{R}$ is the 3×3 rotation matrix, and $\mathbf{t}$ is the 3×1 translation vector.

A 3D predefined landmark on a face mesh is projected onto the 2D image plane by perspective projection, but there usually exists difference between the directly projected 2D positions and the real 2D landmarks. The addition of a 2D displacement (Cao et al. 2014a) can alleviate this drawback.

The transformation from 3D landmarks to 2D landmarks is shown as follows:

$$s_k = \prod(F^{(v_k)}) + d_k, \tag{3.17}$$

where s_k is a 2D landmark in image plane, $F^{(v_k)}$ is its corresponding predefined vertex on 3D face mesh F, d_k is the 2D displacement of s_k, and $\prod$ is the perspective projection operator: For a 3D vertex $p = [X, Y, Z]^T$, its projected position in normalized 2D image plane is $[x = X/Z, y = Y/Z]^T$, and the original 2D landmark is normalized by $[x = (x_o - W/2)/W, y = (y_o - H/2)/W]^T$.

Generally, given L labelled 2D landmarks and corresponding predefined 3D landmarks on its 3D face mesh, the pose matrix $\{\mathbf{R}, \mathbf{t}\}$, identity coefficients u, expression coefficients e, and displacements $\mathbf{D} = \{\mathbf{d_k}\}$ can be solved by minimizing the Huber loss function applied to re-projected error between 2D landmarks and 3D landmarks is shown as follows:

$$arg\min_{p} = \sum_{k=i}^{L} ||d_k||, \tag{3.18}$$

where $\mathbf{P} = \{\mathbf{R}, \mathbf{t}, \mathbf{u}, \mathbf{e}, \mathbf{D}\}$ and d_k is computed on the basis of Eq. 3.18 is shown as follows:

$$d_k = s_k - \prod(\mathbf{R} \times (\mathbf{C_r} \times_{\mathbf{2}} \mathbf{u^T} \times_{\mathbf{3}} \mathbf{e^T}) + \mathbf{t})^{(v_k)}. \tag{3.19}$$

In the case that 2D landmarks has been detected, a nonlinear trust region optimization method like a sparse variant of the LevenbergMarquardt algorithm (Hartley and Zisserman 2003) can be used to solve pose and bilinear parameters, as in Liu et al. (2016, 2017). It is obvious that a reliable landmark detector is necessary, but popular detectors are usually 2D-based and cannot capture large variations of pose and expression out of plane. So we are aiming at solving both 2D landmarks and 3D parameters simultaneously by a cascade regression with shape-indexed features.

Supervised coordinate descent method

In a cascaded regression framework (Xiong and De 2013; Cao et al. 2014c; Xiong and De la Torre 2015), given sufficient training samples, the optimal solution of a nonlinear cost function from a reasonable initialization can be solved by iteratively learning the mappings between current function output and the residual between current input and the optimal solution, as a supervised approximation of the gradient descent method.

As for face alignment and tracking in 2D videos, the nonlinear cost function is based on a feature description like scale-invariant feature transform (Xiong and De 2013), HOG (Feng et al. 2015a) and binary features (Ren et al. 2014) extracted around the landmarks. Given a 2D face image I and L 2D landmarks $\mathbf{s} = [\mathbf{s_1}, \ldots, \mathbf{s_k}, \ldots, \mathbf{s_L}]^\mathbf{T}$, the feature function is denoted as $\mathbf{h}(\mathbf{I}, \mathbf{s})$. On the basis of the bilinear model and 2D displacement model, our input for the function is $\mathbf{P} = \{\mathbf{R}, \mathbf{t}, \mathbf{u}, \mathbf{e}, \mathbf{D}\}$, which generates the 2D landmarks $\mathbf{s}(\mathbf{P})$ by Eq. 3.17, and then the feature function is represented as $h(I, s(\mathbf{P}))$. Denoting $\mathbf{P_0} = \{\mathbf{R_0}, \mathbf{t_0}, \mathbf{u_0}, \mathbf{e_0}, \mathbf{D_0}\}$ as the initialization, and $\mathbf{P}_* = \{\mathbf{R}_*, \mathbf{t}_*, \mathbf{u}_*, \mathbf{e}_*, \mathbf{D}_*\}$ as the optimal solution, the learning object is the residual $\Delta\mathbf{P} = \mathbf{P}_* - \mathbf{P_0} = \{\mathbf{\Delta R}, \mathbf{\Delta t}, \mathbf{\Delta u}, \mathbf{\Delta e}, \mathbf{\Delta D}\}$. Our cost function is defined by minimizing the distance between features of predicted parameters and features of real parameters, as shown in Eq. 3.20:

$$f(\mathbf{P}_0 + \Delta\mathbf{P}) = ||\mathbf{h}(\mathbf{I}, \mathbf{s}(\mathbf{P})_\mathbf{0} + \mathbf{\Delta P})) - \mathbf{h}(\mathbf{I}, \mathbf{s}(\mathbf{P}_*))|| \tag{3.20}$$

A standard cascaded regression for the whole parameters is described in Eq. 3.21:

$$\mathbf{P_{k+1}} = \mathbf{P_k} + \mathbf{W_k}\mathbf{h}(\mathbf{I}, \mathbf{s}(\mathbf{P_k})) + \mathbf{b_k}, \tag{3.21}$$

where $\mathbf{W_k}$ is the learned mapping between the residual $\mathbf{\Delta P_k} = \mathbf{P}_* - \mathbf{P_k}$ and the current shape-indexed feature $\mathbf{h}(\mathbf{I}, \mathbf{s}(\mathbf{P_k}))$ on a training dataset and b_k is the learned bias. However, it is sensitive to changes out of plane and probably

drift away from the real solution, because taking different types of parameters as whole can lead to the cross impact.

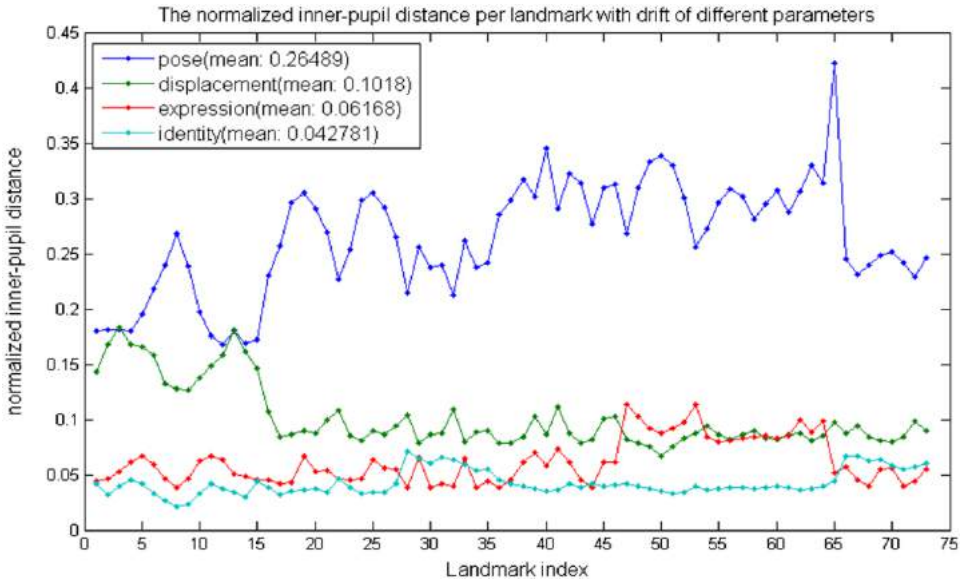


Figure 3.6: *This graph shows the normalized inner-pupil distance per landmark with drift of different parameters, which shows that the pose parameter contributes the most to the landmark update.*

It is known that different parts of $\Delta\mathbf{P}$ make different contributions to the movement of 2D landmarks $\mathbf{\Delta s} = \mathrm{s}(\mathbf{P_0} + \mathbf{\Delta P})\mathrm{s}(\mathbf{P_0})$: Pose residual $\{\mathbf{\Delta R}, \mathbf{\Delta t}\}$ produces large-scale global movement; displacement residual $\Delta\mathbf{D}$ produces large-scale movements of contour landmarks and small-scale inner landmarks; expression residual $\Delta\mathbf{e}$ produces large-scale local movements of parts of landmarks; identity residual $\Delta\mathbf{u}$ produces small-scale global movement. We have evaluated the different movements caused by different parameters on three labeled face databases (Cao et al. 2014b; Huang et al. 2008; Tarrés and Rama 2012). For each type of parameter, we fix others and replace it with a mean value; then, we generate its 2D landmarks via our bilinear model and 2D displacement model. Following, we compute the normalized inter-pupil distance between landmarks generated by real parameters and the replaced ones. As shown in Fig. 3.6, the errors decrease from pose to identity. An example of movements by different parameters also illustrates different-level contributions of different parameters. If a cost function is differentiable, it will be a wise choice to solve different parts one by one with a coordinate decent method. By denoting $\mathbf{M} = \{\mathbf{R}, \mathbf{t}\}, \mathbf{P} = \{\mathbf{M}, \mathbf{D}, \mathbf{e}, \mathbf{u}\}$, an

ideal format of the coordinate decent method for Eq. 3.21 should be:

$$\begin{aligned}
\mathbf{M_{k+1}} &= \mathbf{M_k} - \alpha \mathbf{J}_{\mathbf{h,M}}^{\mathbf{T}}(\mathbf{\Phi}_{\mathbf{M_k}} - \mathbf{\Phi}_*) \\
\mathbf{D_{k+1}} &= \mathbf{D_k} - \alpha \mathbf{J}_{\mathbf{h,D}}^{\mathbf{T}}(\mathbf{\Phi}_{\mathbf{D_k}} - \mathbf{\Phi}_*) \\
\mathbf{e_{k+1}} &= \mathbf{e_k} - \alpha \mathbf{J}_{\mathbf{h,e}}^{\mathbf{T}}(\mathbf{\Phi}_{\mathbf{e_k}} - \mathbf{\Phi}_*) \\
\mathbf{u_{k+1}} &= \mathbf{u_k} - \alpha \mathbf{J}_{\mathbf{h,u}}^{\mathbf{T}}(\mathbf{\Phi}_{\mathbf{u_k}} - \mathbf{\Phi}_*),
\end{aligned} \tag{3.22}$$

where α is the learning ratio, $\mathbf{J_{h,M}}, \mathbf{J_{h,D}}, \mathbf{J_{h,e}}$, and $\mathbf{J_h}, \mathbf{u}$ are Jacobi matrices of different parameters with chain rule by $\mathbf{h}, \mathbf{\Phi}_* = \mathbf{h}(\mathbf{I}, \mathbf{s}(\mathbf{P}_*))$ is the real feature, $\mathbf{\Phi}_{\mathbf{M_k}}$ is the feature obtained by $\{\mathbf{M_k}, \mathbf{D_k}, \mathbf{e_k}, \mathbf{u_k}\}$, $\mathbf{\Phi}_{\mathbf{D_k}}$ is obtained by $\{\mathbf{M_{k+1}}, \mathbf{D_k}, \mathbf{e_k}, \mathbf{u_k}\}$, $\mathbf{\Phi}_{\mathbf{e_k}}$ is obtained by $\{\mathbf{M_{k+1}}, \mathbf{D_{k+1}}, \mathbf{e_k}, \mathbf{u_k}\}$, and $\mathbf{\Phi}_{\mathbf{u_k}}$ is obtained by $\{\mathbf{M_{k+1}}, \mathbf{D_{k+1}}, \mathbf{e_{k+1}}, \mathbf{u_k}\}$.

Because the function h is not differentiable, the analytic solutions of Jacobi matrices cannot be computed directly. Alternatively, the decent mappings $\mathbf{W}_{\mathbf{M_k}} \approx \mathbf{J}_{\mathbf{h,M}}^{\mathbf{T}}$, $\mathbf{W}_{\mathbf{D_k}} \approx \mathbf{J}_{\mathbf{h,D}}^{\mathbf{T}}$, $\mathbf{W}_{\mathbf{e_k}} \approx \mathbf{J}_{\mathbf{h,e}}^{\mathbf{T}}$, and $\mathbf{W}_{\mathbf{u_k}} \approx \mathbf{J}_{\mathbf{h,u}}^{\mathbf{T}}$ between different parameters and function output can be learned as approximation of Jacobi matrices when a sufficient number of training samples are provided. It is noticed that a real feature is known during training but unknown for testing, so it is replaced with the mean feature $\bar{\mathbf{\Phi}}_*$ of all real features for training. Thus, our supervised coordinate descent method is described as:

$$\begin{aligned}
\mathbf{M_{k+1}} &= \mathbf{M_k} - \alpha \mathbf{W}_{\mathbf{h,M}}^{\mathbf{T}}(\mathbf{\Phi}_{\mathbf{M_k}} - \bar{\mathbf{\Phi}}_*) \\
\mathbf{D_{k+1}} &= \mathbf{D_k} - \alpha \mathbf{W}_{\mathbf{h,D}}^{\mathbf{T}}(\mathbf{\Phi}_{\mathbf{D_k}} - \bar{\mathbf{\Phi}}_*) \\
\mathbf{e_{k+1}} &= \mathbf{e_k} - \alpha \mathbf{W}_{\mathbf{h,e}}^{\mathbf{T}}(\mathbf{\Phi}_{\mathbf{e_k}} - \bar{\mathbf{\Phi}}_*) \\
\mathbf{u_{k+1}} &= \mathbf{u_k} - \alpha \mathbf{W}_{\mathbf{h,u}}^{\mathbf{T}}(\mathbf{\Phi}_{\mathbf{u_k}} - \bar{\mathbf{\Phi}}_*).
\end{aligned} \tag{3.23}$$

At the kth cascaded step, the coordinate decent mappings $\mathbb{W}_{\mathbb{M}\neg}$, $\mathbb{W}_{\mathbb{D}\neg}$, $\mathbb{W}_{\neg}$ and $\mathbb{W}_{\approxeq\neg}$ are learned with training pairs $\mathbf{S}_{\mathbf{M_k}} = \{\mathbf{\Delta M_k^i}, \mathbf{\Delta \Phi}_{\mathbf{M_k}}^{\mathbf{i}}\}$, $\mathbf{S}_{\mathbf{D_k}} = \{\mathbf{\Delta D_k^i}, \mathbf{\Delta \Phi}_{\mathbf{D_k}}^{\mathbf{i}}\}$, $\mathbf{S}_{\mathbf{e_k}} = \{\mathbf{\Delta e_k^i}, \mathbf{\Delta \Phi}_{\mathbf{e_k}}^{\mathbf{i}}\}$ and $\mathbf{S}_{\mathbf{u_k}} = \{\mathbf{\Delta u_k^i}, \mathbf{\Delta \Phi}_{\mathbf{u_k}}^{\mathbf{i}}\}$, respectively. Each part is

learned by a linear regression as:

$$\begin{aligned}
\mathbf{W_{M_k}} &= \mathbf{arg}\min_{\mathbf{W_{M_K}}} \sum ||\mathbf{\Delta M_k^I} - \mathbf{W_{M_k \Delta \Phi^i_{M_k}}}|| \\
\mathbf{W_{D_k}} &= \mathbf{arg}\min_{\mathbf{W_{D_K}}} \sum ||\mathbf{\Delta D_k^I} - \mathbf{W_{D_k \Delta \Phi^i_{D_k}}}|| \\
\mathbf{W_{e_k}} &= \mathbf{arg}\min_{\mathbf{W_{e_K}}} \sum ||\mathbf{\Delta e_k^I} - \mathbf{W_{e_k \Delta \Phi^i_{e_k}}}|| \\
\mathbf{W_{u_k}} &= \mathbf{arg}\min_{\mathbf{W_{u_K}}} \sum ||\mathbf{\Delta u_k^I} - \mathbf{W_{u_k \Delta \Phi^i_{u_k}}}||
\end{aligned} \tag{3.24}$$

,

where $\mathbf{\Delta M_k^i} = \mathbf{M_*^i} - \mathbf{M_k^i}$, $\mathbf{\Delta \Phi^i_{M_k}} = \mathbf{\bar{\Phi}_*} - \mathbf{\Phi^i_{M_k}}$, $\mathbf{\Delta D_k^i} = \mathbf{D_*^i} - \mathbf{D_k^i}$, $\mathbf{\Delta \Phi^i_{D_k}} = \mathbf{\bar{\Phi}_*} - \mathbf{\Phi^i_{D_k}}$, $\mathbf{\Delta e_k^i} = \mathbf{e_*^i} - \mathbf{e_k^i}$, $\mathbf{\Delta \Phi^i_{e_k}} = \mathbf{\bar{\Phi}_*} - \mathbf{\Phi^i_{e_k}}$, and $\mathbf{\Delta u_k^i} = \mathbf{u_*^i} - \mathbf{u_k^i}$, $\mathbf{\Delta \Phi^i_{u_k}} = \mathbf{\bar{\Phi}_*} - \mathbf{\Phi^i_{u_k}}$.

Supervised coordinate descent method for face tracking evaluation

In this section, we present the details of applying the supervised coordinate descent method (SCDM) to face tracking application. A core tensor is first computed for the bilinear model by N-mode SVD on a public database. Then our SCDM regression model is trained on augmented public datasets according to Section 3. Finally, the trained SCDM is used for tracking face landmarks in face image sequences.

We use 3D part of Facewarehouse database (Cao et al. 2014b) to build the bilinear model. This database consists of 3D meshes of 150 different persons with 47 expressions for each. There are over 11k vertexes on each mesh. The original data is organized as a 11k vertexes $\times$ 150 identities $\times$ 47 expressions tensor. We compress it to a 11k vertexes $\times$ 50 identities $\times$ 25 expressions core tensor by N-mode singular value decomposition (SVD) on two-mode and three-mode. The core tensor is used in the bilinear model.

There are three public datasets used for training our SCDM: Facewarehouse (Cao et al. 2014b), Labelled Faces in the Wild (LFW) (Huang et al. 2008), and GTAV (Tarrés and Rama 2012). There are 5,904 2D images in Facewarehouse, specially including 1,152 images in left pose and 1,152 images in right pose. Different to the 3D part of this database, it includes 150

different persons with 24 expressions in frontal pose, and also 48 persons with 24 expressions in left pose and right pose. 7,258 images from 3,010 persons in the Labelled Faces in the Wild (LFW) are used, and 1,298 images from 44 persons in the GTAV are used. All the images are labelled with 73 facial landmarks and face bounding boxes are detected by a fast face detector (Yan et al. 2013a).

Pose parameters, expression coefficients, identity coefficients, and displacements have not been given in raw databases, so we solve them with labelled 2D landmarks by minimizing the cost function in Eq. 3.19. The optimization algorithm is an efficient gradient descent method as Liu et al. (2016) adopt. Thus, there are totally 14,460 samples prepared for training. According to the work of Yan et al. (2013a) the histogram of oriented gradients (HOG) feature outperforms scale-invariant feature transform and others for shape-indexed feature in a cascaded regression, so HOG is adopted as our feature function.

Training and testing the SCDM regression model

The mean shape $\bar{\mathbf{P}}_*$ of all parameters for training is used as the initialization $\mathbf{P_0}$ at the first training or testing step. At each training step, we learn pose mapping first, and then update current shape-indexed feature with the new pose, and then we learn displacement mapping and update feature with the new displacement. Similar operations are followed by expression and identity according to Eq. 3.23 and 3.24. A few number of iterations are executed until the training error is under a threshold or the number of iterations reaches the maximum. Four iterations are adopted in our application, which are enough for accurate result.

To initialize, the face bounding-box of the the first frame in a user-specific video is scanned by a face detector (Zhang et al. 2007). Then, we use an object tracker(Danelljan et al. 2014) to capture the bounding box of the face in the subsequent frames. After that, we crop a whole frame by the face bounding box and conduct face alignment and tracking with our SCDM

regression. Testing process also starts from the mean shape $\bar{\mathbf{\Phi}}_*$, and then the different parameters and features are updated one by one with learned coordinate descent mappings based on Eq .3.23. Iteration is executed for several times until reaching the maximum. An example of run-time regression at the first iterative step is shown in Fig. 3.7.

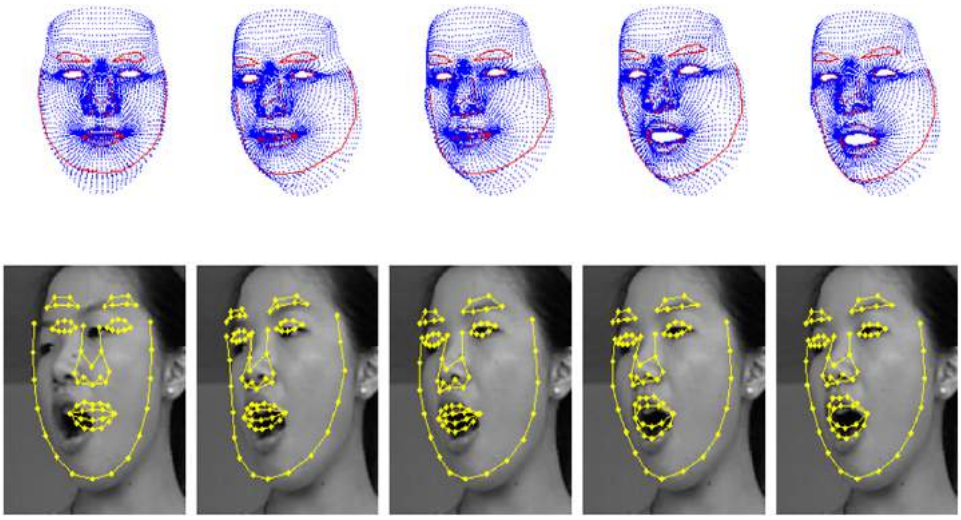


Figure 3.7: *An example of run-time regression at the first iterative step. Starting from the mean shape initialization the learned mapping function update it to output the correct shape for the given image.*

Experimental results

We collected 14 benchmark videos to evaluate the overall performance. The frame number of each video ranges from 190 to 900, and the total number of all the frames is 5135. The resolution of each video is 640×480. All the frames are labelled with 73 2D landmarks. Pose, expression coefficients, identity coefficients, and 2D displacement are also computed on the basis of our 3D bilinear model with labelled 2D landmarks, which are used for reconstruct 3D face meshes of these frames as the ground-truth 3D mesh.

There are different people talking with various expressions, poses, partial occlusions, and illuminations among these sequences. The comparison with state-of-the-art 2D regression-based and model-based methods is performed. 2D regression-based methods include ensemble of regression trees (ERT) (Kazemi and Sullivan 2014), regressing local binary features (Ren

Method	Mesh error	Landmark error
SCDM	**25.1**	**5.1**
3D Robust	41.7	7.2
ERT	92.3	19.2
LBF	88.1	11.4
AAM-1	97.3	21.9
AAM-2	87.9	21.3

Table 3.3: *Landmark tracking error comparison in challenging videos.*

et al. 2014). 2D model-based methods include popular AAM-based methods: AAM-1 (Cootes et al. 2001), AAM-2 (Donner et al. 2006). A recent 3D robust method (Liu et al. 2017) is also compared. These methods are also trained on the same three databases.

We evaluate the 3D mesh errors and 2D landmark errors of our SCDM and other methods. The measure metric of 2D landmark errors is the normalized inner-pupil distance, as used in popular methods (Kazemi and Sullivan 2014; Ren et al. 2014). Because landmark tracking is usually used for capturing 3D facial performance, it is important to get accurate 3D meshes from 2D/3D tracked result. The mesh error is evaluated by calculating the normalized distance between a ground-truth mesh and its reconstructed one. Because the compared 2D-based methods do not directly provide 3D parameters for reconstructing a 3D mesh, we reconstruct their 3D meshes by the following: Solve 3D parameters with the predicted 2D landmarks as we do during training data preparation, and then generate 3D meshes with solved parameters.

The mesh errors and landmark errors of different methods are shown in Table. 3.3. It indicates that our SCDM can track landmarks more accurately. Moreover, 3D meshes directly generated by our SCDM are much more reliable than the reconstructed ones with 2D-based methods. Fig. 3.8 illustrates landmark tracking results in these challenging videos. It can be seen that our SCDM still keep stable localization when the pose and expression change drastically, or partial occlusion occurs. 2D-based methods fail to track in the same situation. More examples of our SCDM tracking in different videos are

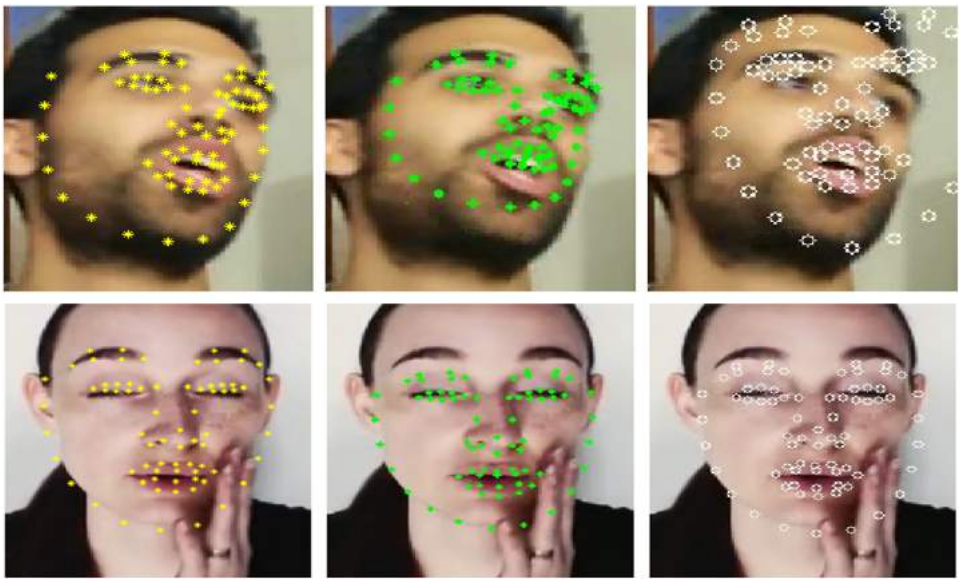


Figure 3.8: *Qualitative comparison to the AAM-based person specific landmark tracking. From left to right: results of our method, AAM-1 and AAM-2. Our results are more robust under partial occlusion and large pose variations.*

shown in Fig. 3.9.

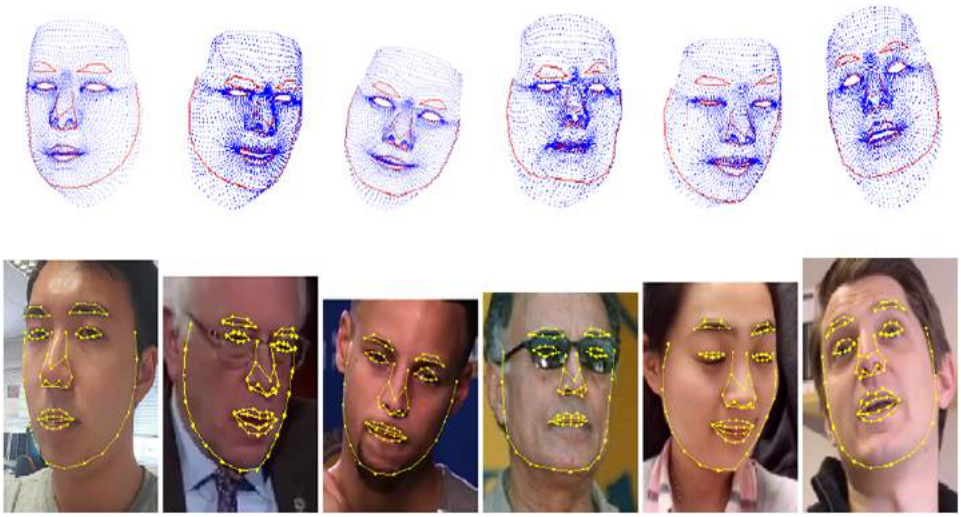


Figure 3.9: *Qualitative SCDM tracking results. Top: Corresponding 3D mesh representation, bottom: 2D landmark projection.*

Summary

In this section, we present a novel data-driven face alignment and tracking method for monocular videos. A bilinear model is used as the 3D prior to strengthen cascaded regression processing. A novel supervised coordinate

descent method is proposed to separately learning the descent mappings between different types of parameters with shape-indexed features individually, which is more stable than the previous whole-parameter regression works. Benefiting from 3D prior of the bilinear model, our method shows more reliable capture ability than popular 2D-based methods while tracking face landmarks with variations out of plane. Our work is easy to be extended for performance-based facial animation and expression transfer in many customized AR/VR applications, which is considered as our future work.

Chapter 4

3D Facial Performance Tracking

Facial performance tracking and editing has attracted attention from both research community and industries for a long time. Among all the research, real time methods in particular are being actively studied due to their effectiveness and many applications. Traditionally, such technologies were only available to institutional users with high-end hardware such as laser scanners and motion capture devices.

Facial performance capture and tracking is a well established research field(Williams 1990; Beeler et al. 2011; Sagar 2006; Ma et al. 2008; Blanz et al. 2003). Traditionally, 3D facial performance capture methods need high-end marker-based motion capture system (Guenter et al. 1998), which tracks a sparse set of markers on a person's face. These kind of methods require the person to wear markers and to be tracked in a specific environment setting, which is both time consuming and invasive. Recently, with the development of stereo vision and depth camera, modern methods (Hernandez et al. 2012; Thies et al. 2015; Hsieh et al. 2015; Weise et al. 2011) no longer require markers and are able to produce superior results under less constraints. Lately consumer-level depth capture devices are becoming more widely available, and state-of-the-art methods (Thies et al. 2015; Hsieh et al. 2015) that use depth information as input have achieved good performance.

Nowadays, with the wide spread availability of web cameras and mo-

bile devices, there has been an increasing demand for real time systems on these low end devices that can track and edit facial performances using only video input to offer instant feedback. As these video-based methods only use 2D images, they are capable of processing both online and prerecorded videos in uncontrolled settings (lighting condition and background, etc). However, because of the inherent lack of depth information, it is an under-constrained problem that requires 3D prior knowledge of the face in order to be solved. Due to the large facial appearance, pose and lighting variation and partial occlusions, existing 3D facial performance tracking methods sometime fail in uncontrolled settings. To this end, we introduce a novel efficient method for tracking and adapting to users appearance online to achieve accurate results, as well as a robust corrective method that densely correct the tracked results.

4.1 Landmark based 3D performance tracking

3D face models are very useful for various applications in computer vision and computer graphics. They provide strong prior of the facial geometry and invariance by modelling the 3D shape of faces and the physical camera imaging process. Although these models are very powerful and useful, their construction is labour intensive. Currently, the most popular publicly available 3D facial datasets are based on Facewarehouse (Cao et al. 2014b) and Basel Face Model (BFM) (Paysan et al. 2009).

The Basel Face Model parametrized the face as triangular meshes with $53,490$ vertices and shared topology, where shape and texture are assumed to be independent from each other, constructing two independent Linear Models as described in Blanz and Vetter (1999). The training dataset consists of face scans of 100 female and 100 male individual of mostly Europeans ethnicity. The age of the persons is between 8 and 62 years with an average of 24.97 years and the weight is between 40 and 123 kilogram with an average of 66.48 kilogram. BFM only models the geometry and texture appearance of the neutral pose of captured individuals, therefore not directly usable for

facial performance tracking. Later some works extended BFM with different expression nodes to achieve facial performance tracking Thies et al. (2016).

Facewarehouse parametrized the face as triangular meshes with 11510 vertices and shared topology. The dataset consists of 150 male and female aged 7 to 80 from various ethnic backgrounds. In addition to the neutral facial expression, including the neutral expression and 19 other expressions such as mouth-opening, smile, kiss, etc. These captures are used to compute 47 different facial expressions for each person by deforming a template mesh. Finally, the dataset is arranged into a 3-mode tensor and compressed via N-mode SVD (Vlasic et al. 2005). Compared with previous 3D facial datasets, for each individual in FaceWarehouse, there is a much richer collection of expressions, which enables a more expressive depiction of most human facial actions.

We use Facewarehouse Cao et al. (2014b) as our 3D mesh database. It consists of the facial geometry of 150 persons, and 47 expressions for each of the said people. This database is accompanied with 2D facial images and their corresponding manually labeled 74 facial landmarks. We train a 2D landmark detector on this dataset. Since all faces shared the same topology we select only the frontal facial vertices and rearrange them into a rank-three (3 mode) data tensor.

The landmark prediction produced by the detector can be noisy at times, in order to generate realistic results we have to regularise the expression coefficients to make sure they are within valid ranges. Uncompressed original data is represented as blendshapes and has semantic meanings such as closing eyes, opening mouth and frowning. However, to reduce the computational cost of fitting new expression and identity, we compress the tensor into a 4k vertices $\times$ 50 identities $\times$ 25 expressions core using the method from Kolda and Sun (2008). Thus, we reconstruct the original expression blendshapes and weights given an identity coefficient, and use them instead of the compressed weights. Original decompressed blendshapes can be reconstructed using the product of the $4000 \times 50 \times 25$ tensor core C and 50×150 U_{id} or

25×47 U_{exp} orthonormal matrices.

$$B_{exp} = C \times_2 U_{id} \tag{4.1}$$

$$B_{id} = C \times_3 U_{exp} \tag{4.2}$$

B_{exp} is a person with different facial expressions, B_{id} is the same expression performed by different individuals. When solving for identity we always use the compressed core with early stopping to prevent over-fitting, while transferring expression and animating target characters we solve with the reconstructed blendshapes, with early stop and clamping as a regularisation to generate plausible results.

4.1.1 3D performance tracking from 2D landmarks

3D pose estimation

After we localise the facial landmarks using a 2D landmark detector, we rigidly align the 3D mesh to the 2D landmarks. 3D coordinates of the inner landmarks such as the ones on nose, eyes, mouth and eyebrow are located using fixed indices. However the vertex indices on the face contour have to be updated in regard to different poses and expressions. We can find the contour indices by projecting the vertices onto a plane and sampling its convex hull uniformly. To reduce computational cost we only project a predefined set of vertices. We project a predefined set of vertices - which are organised into horizontal lines in a clockwise order - onto a plane and compute their convex hull. Next we find the closest point to its first and last points. Finally, we connect the two points with uniformly sampled points on the convex hull, and the indices are copied from the closest vertices to these points, as shown in Fig. 4.1.

After the indices have been found we estimate the pose by minimising distance between its projection and 2D landmarks. The initial estimation

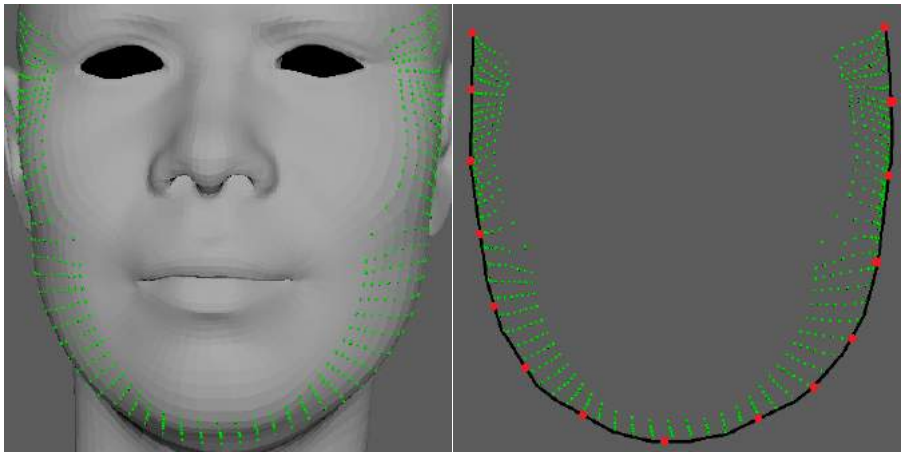


Figure 4.1: *Convex hull based contour facial landmark sampling. A predefined set of vertices are projected to the image space according to the pose and camera parameters. The convex hull of the projection is computed a the facial landmark indices are sampled from the hull uniformly.*

of the rotation and translation parameters are computed from Direct Linear Transform (DLT) (Abdel-Aziz 1971), which is an algorithm which solves a set of variables from a set of similarity relations. Given the camera model, the standard DLT parameters are formed to reflect the relationships between the object-space reference frame and the image-plane reference frame, which is then refined iteratively via nonlinear optimisation methods such as trust-region (Coleman and Li 1996) or line search (Liu and Nocedal 1989).

First way to express this problem mathematically is to cast it as a linear least squares problem. This approach is known as the DLT approach, and it is interesting because linear least-squares have a closed-form solution which can be found robustly using the Singular Value Decomposition. However, this approach assumes that the camera pose has 12 degrees of freedom (DOF) when really it has only 6 (3 for the 3D rotation plus 3 for the 3D translation). To obtain a 6 DOF camera pose from the result of this approach an approximation is needed, which is not covered by the linear cost function of the DLT and leads to an inaccurate solution.

Second way to express the Perspective-n-Point (PnP) problem mathematically is to use the geometric error as a cost function, and to find the camera pose that minimizes the geometric error. Since the geometric error is non-linear, this approach estimates the solution using iterative solvers, such

as the Levenberg Marquardt algorithm. Such algorithms can take into account the 6 degrees of freedom of the camera pose, leading to accurate solutions. However, since they are iterative approaches, they need to be provided with an initial estimate of the solution, which in practice is often obtained using the DLT approach.

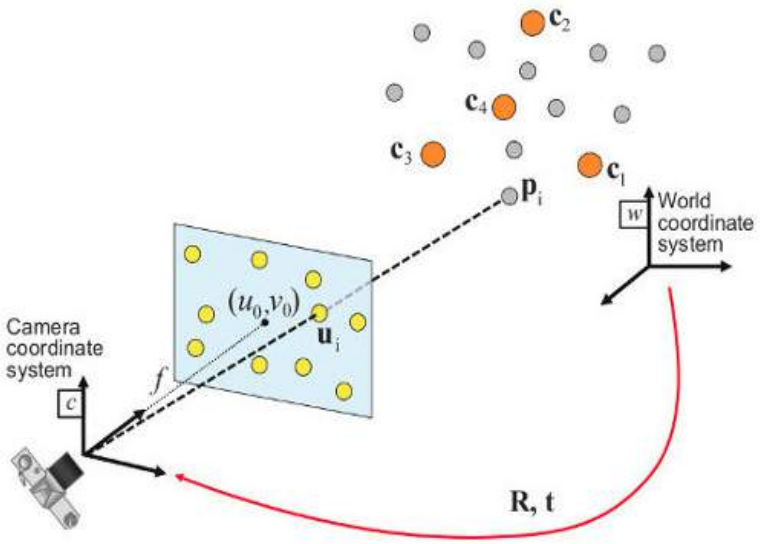


Figure 4.2: *Illustration of the pinhole camera model.*

Suppose a point on the 3D object in its rest pose is located at $\mathbf{X} = [X, Y, Z]$ and its 2D projection in current camera pose is $\mathbf{Y} = [u, v, 1]$. In the ideal pinhole camera model the transformation can be written as Eq. 4.3

(Sturm 2014):

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + t \tag{4.3}$$

$$u = x/z \times f_x + c_x$$

$$v = y/z \times f_y + c_y,$$

where R and t are the rotation and translation parameters, f_x and f_y are the focal lengths and c_x and c_y are the principle points. The DLT parameters L can be obtained using the least square method as in Eq. 4.4 (Abdel-Aziz 1971).

$$\begin{gathered} \mathbf{X} \cdot L = \mathbf{Y} \\ (\mathbf{X}^t \cdot \mathbf{X}) \cdot L = \mathbf{X}^t \cdot \mathbf{Y} \\ (\mathbf{X}^t \cdot \mathbf{X})^{-1} \cdot (\mathbf{X}^t \cdot \mathbf{X}) \cdot L = (\mathbf{X}^t \cdot \mathbf{X})^{-1} \cdot (\mathbf{X}^t \cdot \mathbf{Y}) \\ L = (\mathbf{X}^t \cdot \mathbf{X})^{-1} \cdot (\mathbf{X}^t \cdot \mathbf{Y}). \end{gathered} \tag{4.4}$$

From a set of n point correspondences we now have a $2n \times 12$ matrix A formed by stacking each of the equations from their respective point correspondences. The projection matrix for a given camera can be computed by solving the set of equations $Lp = 0$, where p is a 3×4 projection matrix. In order to solve for L, we obtain the singular value decomposition (SVD) of L and take the smallest singular value as our solution and thus determine the linear transformation (Bardsley and Li).

Next we employ nonlinear optimisation to refine the output from DLT. The group of 3D rotations (SO3) is a matrix lie group. The Jacobian J of the rotation parameter is given by in the work of Taylor and Kriegman (1994);

Curtis (2012), where the rotation matrix is converted to the Rodrigues' exponential form $\mathbf{R}(w) = \mathbf{R_0}\exp\{J(\omega)\} \quad \omega \in \mathbb{R}^3, \sqrt{\omega^t\omega} < \pi$, where $\omega = (\omega_x, \omega_y, \omega_z)^t$ and $J(\omega)$ is the skew symmetric operator $J : \mathbb{R}^3 \to so(3)$ given by:

$$J(\omega) = \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{bmatrix}, \tag{4.5}$$

where the Jacobian is derived by taking the derivative in respect to $[\partial\omega_x, \partial\omega_y, \partial\omega_z]$. Given a set of 3D points and their 2D projection points, the quadratic approximation of the objective function can be constructed as follows:

$$O(R(\omega)) \approx O(R_0) + J^t\omega + \omega^t H\omega = 0, \tag{4.6}$$

where the Hessian is simulated by JJ^t. In the real world facial landmark detection scenario, the 2D projections points observed from the image is often noisy or completely wrong on blurry frames. Huber loss as in Eq. 4.7, for example, is a robust loss term that can substitute the quadratic object function to reduce the influence of outliers.

$$|a|_\delta = L_\delta(a) = \begin{cases} \frac{1}{2}a^2 & \text{for } |a| \le \delta, \\ \delta(|a| - \frac{1}{2}\delta), & \text{otherwise.} \end{cases} \tag{4.7}$$

For 3D facial performance tracking in continuous videos, the temporal coherence could be explored by adding a smoothness term $|\Delta\{R_i, t_i\}|$ that ensure the pose from neighbouring frames varies smoothly. By defining the projection operator of a vertex V on the 3D facial mesh as $\prod$, the objective function can be expressed as Eq. 4.8:

$$\min_{R,t} \sum_{i,j} |\prod(R_i \cdot V_j + t_i) - L_j|_\delta + |\Delta R_i| + |\Delta t_i|, \tag{4.8}$$

which is the sum of ith frames and jth landmarks L.

Expression Estimation

Next we solve for the expression coefficient by fixing R and t. The facial landmark vertices of different expressions can be represented by the product of the 25×1 expression coefficient E and the set of person specific $25 \times 50 \times 15510$ blendshapes B, as in Eq .4.9:

$$V_{xyz} = B \cdot E \tag{4.9}$$

, where V_{xyz} denotes the blended vertices V with three 3×1 coordinates $[x, y, z]$. Given an identity coefficient I, which represent the same individual performing the different expressions, the data term can be written as:

$$|L - \prod(B \cdot E)|_{\delta} \tag{4.10}$$

and save the blended vertices as F_{xyz} and their projection as P which are reused when computing the derivative.

Respectively, derivatives of the two operators with respect to the lth expression coefficient can be written as:

$$(f \cdot P - L)(f \cdot \frac{B_{xy}^{(l)} + B_{z}^{(l)} \cdot P}{F_z}) \tag{4.11}$$

for perspective projection. The energy term described in Eq. 4.11 is the minimum term for solving the expression coefficients, there are variants of it with improved robustness, accuracy or efficiency, which will be described and investigated in following sections.

For certain applications it might be sufficient to use only orthogonal projection as its computational cost is lower, the difference between orthogonal and perspective projection is shown in Fig. 4.3. For commodity cameras with different focal lengths, it is important to use the perspective projection model . The rendering in the third column is generated by slightly rotating

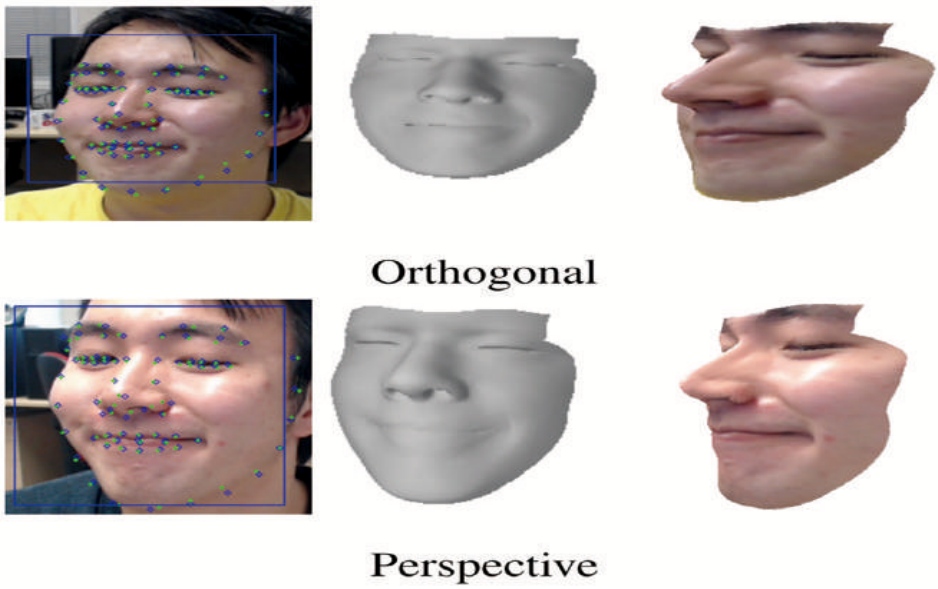


Figure 4.3: *Examples of the orthogonal and perspective projection are shown this figure. Although orthogonal projection is simpler to compute, it cannot model the perspective effect or take the focal length into consideration.*

the tracked face to show depth. Since the orthogonal operator only scales the object, assuming the depth difference is small enough that its appearance will not be greatly affected by focal length and depth, the fitted mesh appears squashed with rotation. Nevertheless, we are able to successfully recover expressive facial expression for both projection operators. Although the two operators are different, the expression coefficients are interchangeable.

For some application such as facial expression transfer, sometimes it is more desirable to use the uncompressed blendshapes. By subtracting the neutral expression from every other expressions the vertices can be represented similarly

$$V_{xyz} = (B - N) \cdot E + N \tag{4.12}$$

where N is the neutral face. The derivative is computed in the same way as Equation (4.11). Compared to compressed tensor, it might be easier to impose constrains on the coefficients of the original blendshapes coefficients as they retains the semantic meaning that is lost in the tensor compression.

Identity Estimation

The identity coefficient can be solved similarly. Given a specific expression, we can create a set of blendshapes which represent different individuals performing the same expression. However, unlike expression coefficient solving identity on a single frame is not sufficient and often leads to unrealistic results. Thus, we solve identity coefficient I using a set of frames with distinctive poses and expressions. We denote the ith blendshapes with jth frame as $B^{(i,j)}$ and minimise the distance

$$\sum_{i,j} |L^j - \prod (B^{i,j)} \cdot I)|_\delta \tag{4.13}$$

while fixing all the other parameters.

It is also unnecessary to solve for the focal length for single frames. Given a set of 2D landmarks and projected vertices, we can solve the focal length by minimising

$$\sum_{j} |\prod (F_{xy}^j) * f_{xy} - L^j|_\delta, \tag{4.14}$$

which can be solved analytically. The routine of updating the focal length and identity coefficient for the online tracking procedure is summarised in Algorithm 1.

Algorithm 1: User Adaptation

```
input : New frame
while not converged do
    if New frame is valid then
        Add frame;
    forall frames do
        Estimate pose;
        Solve Eq. (4.13) for identity of a single frame;
        Solve Eq. (4.11) for expression;
    Solve Eq. (4.13, 4.14) for identity and focal length of all frames;
    Update texture;
output: New identity, focal length and updated texture
```

4.1.2 Monocular Expression Transfer

Monocular expression transfer is an important application for the proposed method. Facial expression transfer has been actively researched in the past few years. Existing methods either suffer from depth ambiguity or require special hardware. In this section we present a novel marker-less, real time facial transfer method that requires only a single video camera. We develop a robust model, which can adapt to user specific facial data. It computes expression variances in real time and rapidly transfers them onto a target character either from images or videos.

Our method can be applied to videos without prior camera calibration and focal adjustment. It enables realistic online facial expression editing and performance transferring in many scenarios such as: video conference; news broadcasting; lip-syncing for song performances; etc. With low computational cost and hardware requirement, our method tracks a single user at an average of 38 fps, and runs smoothly even in web browsers. We present a real time facial expression transfer method with a single RGB camera. Our approach requires low computational resource and is robust in uncontrolled environments.

System Overview

Our method instead deals with transferring expression onto a target character from images or videos, where previous approaches are either not applicable or rely on depth information Thies et al. (2015). We need track the facial mesh of the actor and target as well as modelling their appearances, which is difficult to achieve real time performance.

The overview of our method is illustrated in Fig.4.4, note that we keep the eye region of the target unchanged. To initialise, both images of the actor and the transfer target are first scanned by a face detector (Zhang et al. 2007). We use an object tracker (Danelljan et al. 2014) to get the bounding box of the face. Then, we use a 2D landmark detector to localise the facial

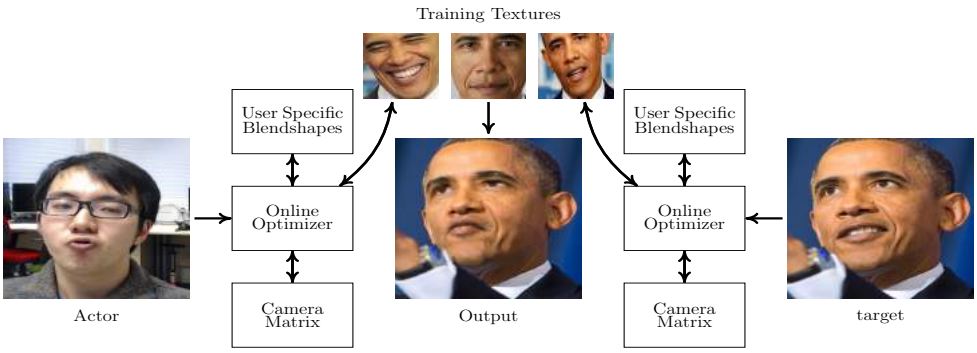


Figure 4.4: *Overview of our method*

landmarks from within the bounding box of the detected face. Next, we solve the pose, expression coefficient and camera matrix by minimizing difference between the projection of their combination and the 2D landmarks using an average identity. After a sufficient number of frames are processed we solve for the user specific identity coefficients. This process can be repeated many times if necessary during the whole runtime. We model the transfer target's appearance using all images available and solve for its pose, expressions and identity. Finally, given the actor's expression we generate appropriate texture for the target and we render the target's face with the actor's expression.

User Adaptation

We regularly select distinctive frames to solve for users' identity coefficients. The first few initial expressions, poses and the flattened landmark vectors are concatenated together to form a parameter base matrix A. When a new frame is received, we try to use this base to reconstruct the solved parameters. If the reconstruction error is larger than a threshold, the frame is added to the base. Once there are enough newly added frames we recompute the identity coefficient focal length, which can be repeated multiple times during runtime if necessary.

We use our learned appearance model to quickly verify whether a frame is valid by thresholding the sum of pixel differences of the extracted texture to the texture produced from the learned appearance model. After a few frames the base size becomes stable and most of the possible combinations of users'

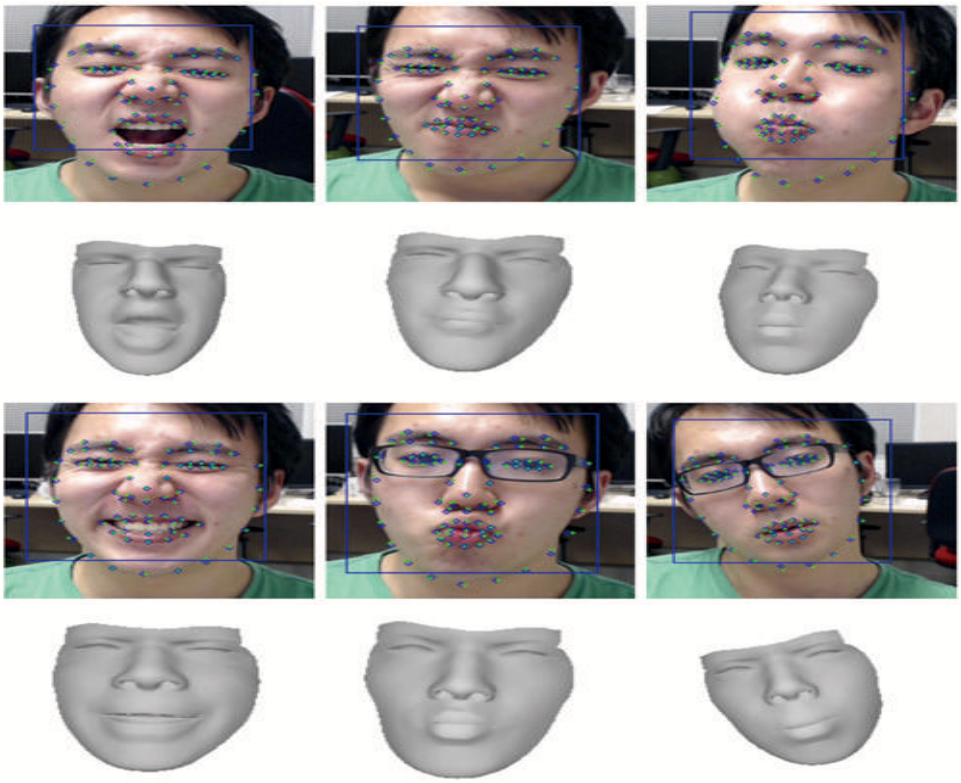


Figure 4.5: *Example qualitative tracking results shows the tracked mesh on the second row corresponding to the first row, where faces are detected, facial landmark are detected and 3D model is fitted to the 2D facial landmarks.*

expression and pose will be collected.

Appearance Learning

After the identity coefficients and corresponding blendshapes have been computed, we can directly transfer low frequency information from an actor to a target. However, details such as wrinkles cannot be transferred because this high frequency information is not captured by the 2D facial landmarks.

The original UV coordinates of the models in Facewarehouse include the whole head. We crop the frontal facial UV coordinates and use the holes formed by the eyeballs and inner mouth to place our UV coordinates for transferring the eyes and inner mouth texture. Texture is extracted by rendering the UV coordinates as the vertices and their corresponding vertex projection as UV coordinates. This helps us to normalise different textures to the same format, which makes learning possible. Since the structure of our UV map is

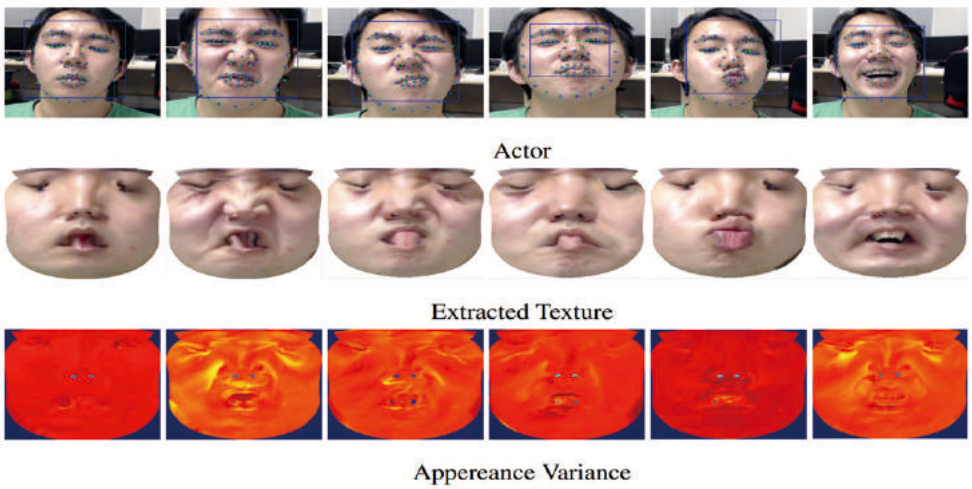


Figure 4.6: *Given a series of image frames and their corresponding tracked meshes, textures can be extracted via back projection, and the person specific appearance is modelled to capture texture variation under different facial expressions.*

the same for different persons and expressions, we can transfer our the new texture to different targets.

After the 3D mesh projection is matched to its corresponding image, we use the texture map as the vertex position and the projected vertex of the mesh to extract a texture. We categorise the extracted textures by their corresponding expressions. The appearance model is updated by finding the most similar expression from the collected parameter base and using the new texture, which is passed through a low-pass filter to avoid outliers and exponential smoothing to void outliers.

Finally, we compute an average texture from all of the collected textures. Each texture can be represented by a linear combination of its differences from the average texture

Texture Generation

When transferring the expression of the actor to the target, we need to generate appropriate texture for the actor's expression. Given a learned appearance texture variance base, we compute a vector that contains the distance between the given expression and the expressions from parameter base. We normalise

this vector and use it as a coefficient to generate a texture, which is a linear combination of the texture base. We overlay it onto the texture of the new frame, which contains high frequency details and makes the result realistic. To keep the computational cost low, we need to keep the length of the texture coefficient short. Thus, instead of using dimension reduction methods such as Principle Component Analysis (Kim et al. 2002), we omit textures with low variance from the average texture and keep the expression indices unchanged, which allows us to deliver real time performance.

The generated texture can also be used to validate whether the parameters estimated for the current frame are valid by checking its sum of pixel differences to the extracted texture. We reject inaccurate frames from the texture updating, identity and focal length solving process. The texture variance base is shown in Fig. 4.6, where the first row is the input/query frames, the second row is the extracted textures and the third row is their variances from the base texture coloured in JET colormap. Note that the colour intensity of the third row between each frame does not necessarily reflect its real value as we normalise them individually instead of jointly to make regions with different high frequency details highlighted.

4.1.3 Summary

To test the efficiency and accuracy of our online method, we compared our results with offline results from the identity, texture and focal length perspectives. By treating the offline results as the ground truth, we calculated the errors from our online method, which converged to good result within 300 frames (10 seconds) as shown in Fig. 4.8.

The tracked mesh is shown in Fig.4.5, which demonstrates that our system can robustly track the rough movement and expression. Although high frequency details are missing, combined with the texture generated from our learned appearance model we can still deliver realistic results as shown in Fig. 4.7.

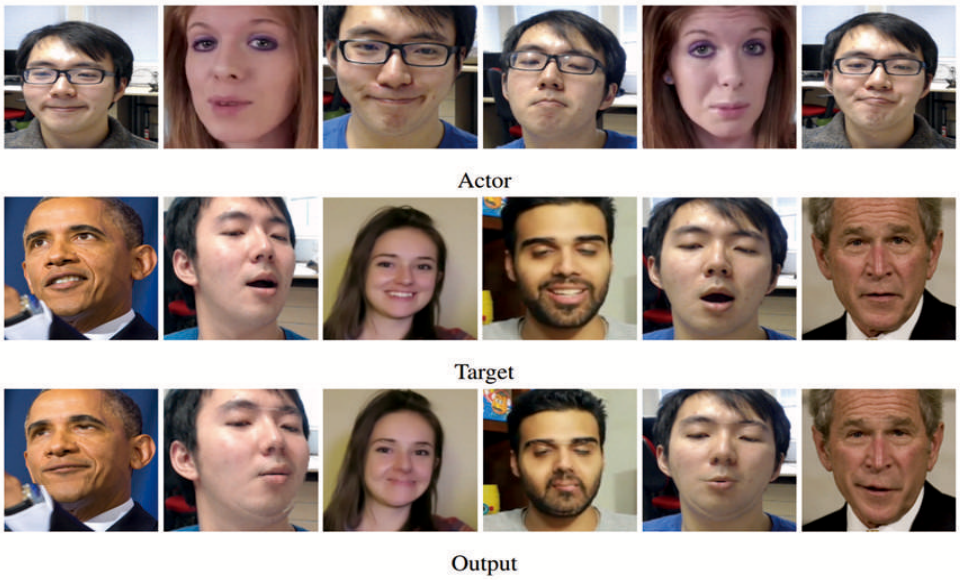


Figure 4.7: *Qualitative facial expression transfer results. The facial expression of the actor from the first row is transferred to the target in the second row, and the final render results composited from these two source of input are shown in the last row.*

Compared to state of the facial mesh tracking method Cao et al. (2014a), in which a regressor is trained to produce expression coefficient and for each frame it is solved again for current frame to improve expressiveness, our method instead directly solve expression coefficient for each frame. We smoothed the solved results with double exponential smoothing to take into account the expression trend in video sequence, which is simpler than cross frame regularization. Our method is thus more efficient. On similar hardware, it achieved an average of 38 fps compared to 28 fps reported in Cao et al. (2014a).

We have introduced a novel real time facial expression transfer method using a single video camera. Experiments show that our method is able to effectively transfer the facial expressions between source and target actors. It provides facial expression transfer and also can be used to drive virtual characters, and the appearance learning scheme we proposed allows us to generate high frequency details on the target.

Our facial mesh tracking pipeline is robust against occlusion and illumination changes. Even if it fails in extreme situations, we can redetect and

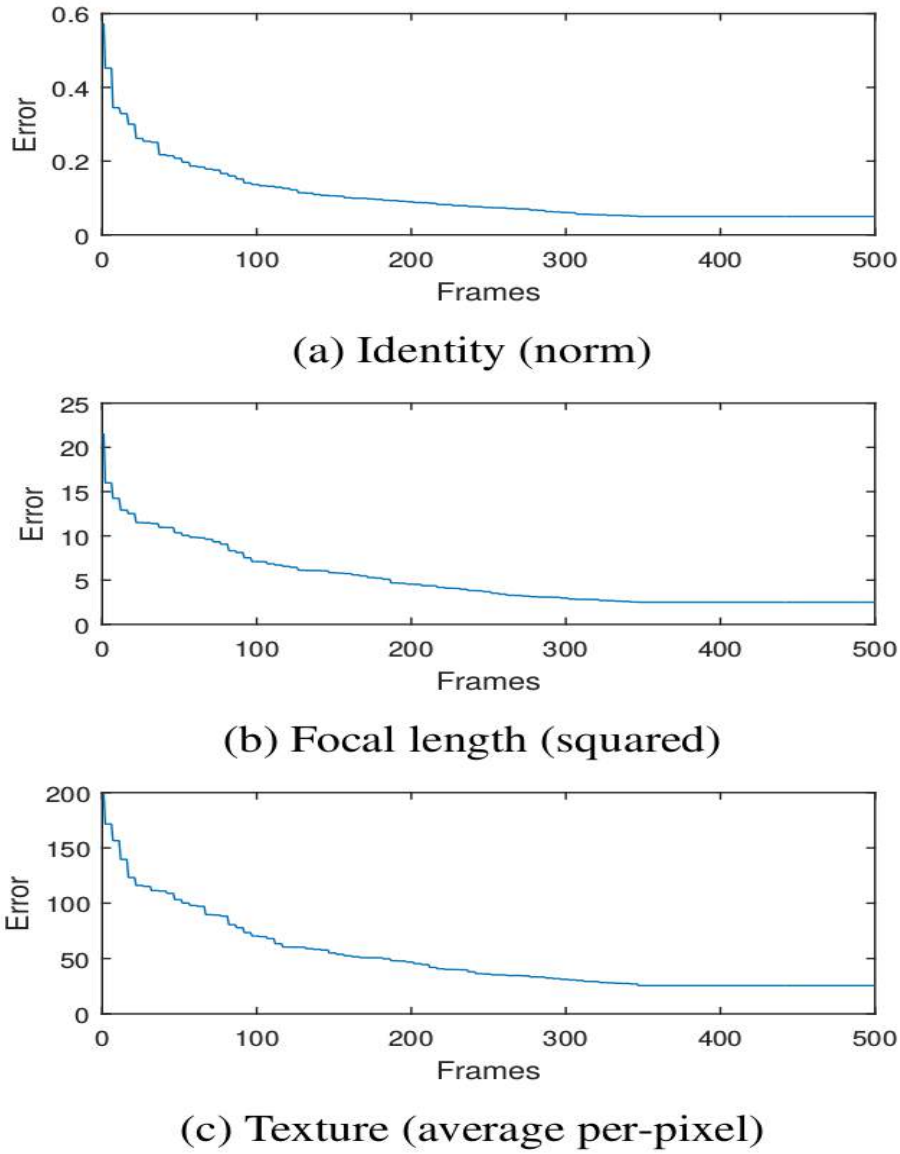


Figure 4.8: *The tracked identity, focal length and texture parameters of our online tracking methods are compared to the results computed offline with all of available frames, and their differences are shown as L2 norms in the corresponding graphs.*

quickly recover since our appearance model rejects the outliers from contaminating the long term identity, focal length and expression specific textures solving. Although we have accounted for possible noise and brief partial occlusion, our method cannot deal with long term partial occlusion and gadgets such as glasses.

With our demo we have already set up a live demo demonstrating that our mesh tracking method is efficient enough to run smoothly on web browsers. For future work, we will save the learned model on client machines as a live puppetry and transfer only expression coefficient to greatly reduce bandwidth requirement for real time video conference. We will also use traditional facial motion capture device to build a ground truth dataset to objectively validate our method.

4.2 Robust Facial Tracking

In this section, we present a novel approach for automatically detecting and tracking facial landmarks across poses and expressions from in-the-wild monocular video data, e.g., YouTube videos and smartphone recordings. Our method does not require any calibration or manual adjustment for new individual input videos or actors.

Firstly, we propose a method of robust 2D facial landmark detection across poses, by combining shape-face canonical-correlation analysis with a global supervised descent method. Since 2D regression-based methods are sensitive to unstable initialization, and the temporal and spatial coherence of videos is ignored, we utilize a coarse-to dense 3D facial expression reconstruction method to refine the 2D landmarks. On one side, we employ an in-the-wild method to extract the coarse reconstruction result and its corresponding texture using the detected sparse facial landmarks, followed by robust pose, expression, and identity estimation. On the other side, to obtain dense reconstruction results, we give a face tracking flow method that corrects coarse reconstruction results and tracks weakly textured areas; this is used to

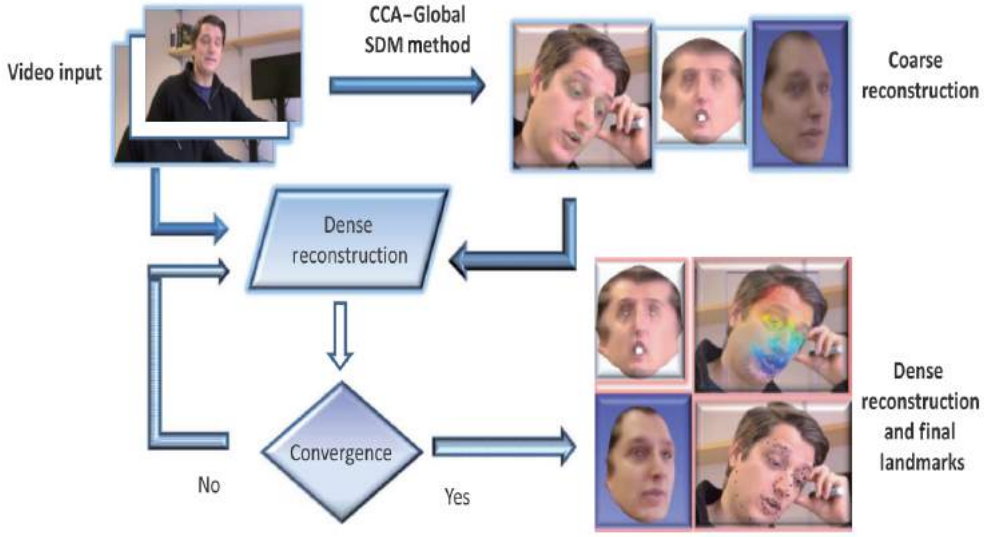


Figure 4.9: *Flowchart overview of the robust dense tracking pipeline. After the appearance model is initialized from coarse landmark reconstruction, we employ a robust optical flow based method to correct misaligned frames and generate a new more accurate appearance model and iterate until convergence.*

iteratively update the coarse face model. Finally, a dense reconstruction result is estimated after it converges. Extensive experiments on a variety of video sequences recorded by ourselves or downloaded from YouTube show the results of facial landmark detection and tracking under various lighting conditions, for various head poses and facial expressions. The overall performance and a comparison with state-of-art methods demonstrate the robustness and effectiveness of our method.

To this end, we have designed a new ITW facial landmark detection and tracking method that employs optical flow to enhance the expressiveness of captured facial landmarks. A flowchart of our work is shown in Fig. 4.9. First, we use a robust 2D facial landmark detection method which combines canonical correlation analysis (CCA) with a global supervised descent method (SDM). Then we improve the stability and accuracy of the landmarks by reconstructing 3D face geometry in a coarse to dense manner. We employ an ITW method to extract a coarse reconstruction and corresponding texture via sparse landmark detection, identity, and expression estimation. Then, we use a face tracking flow method that exploits the coarsely reconstructed model

to correct inaccurate tracking and recover details of the weakly textured area, which is used to iteratively update the face model. Finally, after convergence, a dense reconstruction is estimated, thus boosting the tracked landmark result. Our contributions are three fold:

- A novel 2D facial landmark detection method which works robustly across a range of poses, based on combining shape-face CCA with SDM.
- A novel 3D facial optical flow tracking method for robustly tracking expressive facial landmarks to enhance the location result.

4.2.1 Limitation of existing methods

As a key part in facial performance capture, robust facial landmark detection across poses is still a hard problem. Typical generative models including active shape models (Cootes et al. 1995), active appearance models (Cootes et al. 1998), and their extensions (Cristinacce and Cootes 2006; Gonzalez-Mora et al. 2007; Lee and Kim 2009) mitigate the influence of illumination and pose, but tend to fail when used in the wild. Recently, discriminative models have shown promising performance for robust facial landmark detection, represented by cascaded regression-based methods, e.g., explicit shape regression (Cao et al. 2014c), and the supervised descent method (Xiong and De 2013).

To reconstruct the 3D geometry of the face, existing methods can be grouped into two categories. One group aims to robustly deliver coarse results, while the other one aims to recover fine-grained details. For example, methods such as those in (Suwajanakorn et al. 2014; Garrido et al. 2013; Shi et al. 2014) can reconstruct details such as wrinkles, and track subtle facial movements, but are affected by shadows and occlusions. Robust methods such as (Cao et al. 2015, 2016; Saito et al. 2016) can track facial performance in the presence of noise but often miss subtle details such as small eyelid and mouth movements, which are important in conveying the targets emotion

and to generate convincing animation. Although we use a 3D optical flow approach similar to the work of Suwajanakorn et al. (2014) to track facial performance, we also deliver stable results even in noisy situations or when the quality of the automatically reconstructed coarse model is poor.

Many recent works following the cascaded regression framework consider how to improve efficiency (Xing et al. 2014; Yan et al. 2013a) and accuracy, taking into account variations in pose, expression, lighting, and partial occlusion (Burgos-Artizzu et al. 2013; Yang et al. 2015a). Although previous works have produced remarkable results on nearly frontal facial landmark detection, it is still not easy to locate landmarks across a large range of poses under uncontrolled conditions. A few recent works (Feng et al. 2015b; Yang et al. 2015b; Zhu et al. 2015) have started to consider multi-pose landmark detection, and can deal with small variations in pose. How to solve the local minima issue caused by large differences in pose is our concern.

On the other hand, facial landmark detection and tracking can benefit from reconstructed 3D face geometry based on existing 3D facial expression databases. Remarkably, Cao et al. (2014a) extended the 3D dynamic expression model to work with even monocular video, with improved performance of facial landmark detection and tracking. Their methods work well with indoor videos for a range of expressions, but tend to fail for videos captured in the wild (ITW) due to uncontrollable lighting, varying backgrounds, and partial occlusions. Many researchers have made great efforts on dealing with ITW situations and have achieved many successes (Cao et al. 2014a; Liu et al. 2016; Tzimiropoulos and Pantic 2013).

However, the expressiveness of captured facial landmarks from these ITW approaches is limited since most pay little attention to very useful details not represented by sparse landmarks. Additionally, optical flow methods have been applied to track facial landmarks (Suwajanakorn et al. 2014). Such a method can take advantage of fine-grained detail, down to pixel level. However, it is sensitive to shadows, light variations, and occlusion, which makes it difficult to apply in noisy uncontrolled environments.

Since the input is uncontrolled in ITW videos, person specific facial landmark detection methods such as AAM are inappropriate. AAM methods explicitly minimize the difference between the synthesized face image and the real image, and are able to produce stable landmark detection results for videos in controlled environments. However, conventional wisdom states that their inherent facial texture appearance models are not powerful enough for ITW problems. Although in recent literature (Tzimiropoulos and Pantic 2013) efforts have been made to address this problem, superior results to other ITW methods have not been achieved. Regressor-based methods, on the other hand, work well in the face of ITW problems and are robust (Burgos-Artizzu et al. 2013), efficient (Ren et al. 2014), and accurate (Cao et al. 2014c; Cootes et al. 2012).

Most ITW landmark detection methods were originally designed for processing single images instead of videos (Xiong and De 2013; Cao et al. 2014c; Kazemi and Sullivan 2014). On image facial landmark detection datasets such as 300-W (Sagonas et al. 2013a), Helen (Zhou et al. 2013), and LFW (Huang et al. 2007), existing ITW methods have achieved varying levels of success. Although they provide accurate landmarks for individual images, they do not produce temporally or spatially coherent results because they are sensitive to the bounding box provided by face detector. ITW methods can only produce semantically correct but inconsistent landmarks, and while these facial landmarks might seem accurate when examined individually, they are poor in weakly textured areas such as around the face contour or where a higher level of detail is required to generate convincing animation. One could use sequence smoothing techniques as post processing (Cao et al. 2014a; Liu et al. 2016), but this can lead to an oversmoothed sequence with a loss of facial performance expressiveness and detail.

It is only recently that an ITW video dataset (Shen et al. 2015) was introduced to benchmark landmark detection in continuous ITW videos. Nevertheless, the number of facial landmarks defined in (Shen et al. 2015) is limited and does not allow us to reconstruct the person's nose and eyebrow shape. Since we aim to robustly locate facial landmarks from ITW videos,

we collected a new dataset by downloading YouTube videos and recording video with smartphones, as a basis for comparing our method to other existing methods.

In order to accurately track facial landmarks, it is important to first reconstruct face geometry. Due to the lack of depth information in images and videos, most methods rely on blendshape prior to model nonrigid deformation while structure-from-motion, photometric stereo, or other methods (Furukawa and Ponce 2009) are used to account for unseen variation (Cao et al. 2016; Ichim et al. 2015) or details (Suwajanakorn et al. 2014; Garrido et al. 2013). Due to the nonrigidness of the face and depth ambiguity in 2D images, 3D facial priors are often needed for initializing 3D poses and to provide regularization. Nowadays consumer grade depth sensors such as Kinect have been proven successful, and many methods (Cao et al. 2014b; Weise et al. 2011) have been introduced to refine its noisy output and generate high quality facial scans of the kind which used to require high end devices such as laser scanners (Blanz and Vetter 1999). In this work we use the FaceWarehouse (Cao et al. 2014b) as our 3D facial prior.

4.2.2 Coarse landmark detection and reconstruction

An example of coarse landmark detection and reconstruction is shown in Fig. 4.10. To initialize our method, we build an average shape model from the input video. First, we run a face detector (Yan et al. 2013b) on the input video to be tracked. Due to the uncontrolled nature of the input video, it might fail in challenging frames. In addition to filtering out failed frames, we also detect the blurriness of remaining ones by thresholding the standard deviation of their Laplacian filtered results. Failed and blurry frames are not used in coarse reconstruction as they can contaminate the reconstructed average shape. We employ the robust face landmark detector described in the previous Chapter.

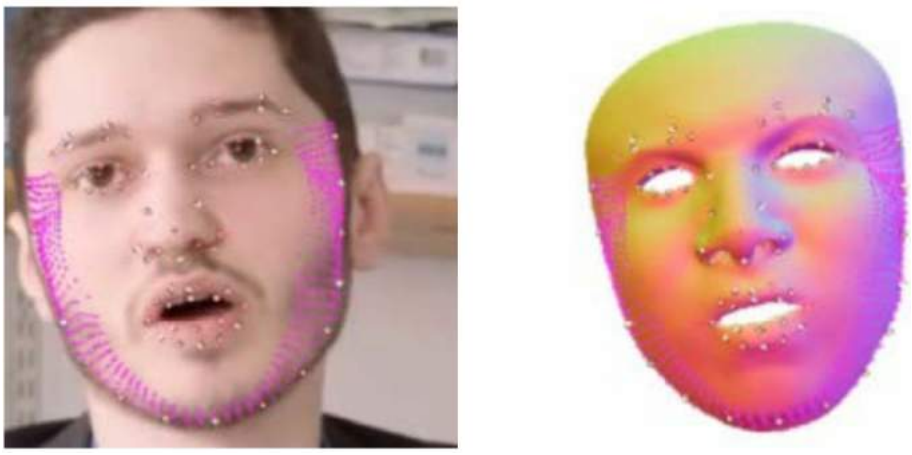


Figure 4.10: *Illustration of the 2D landmark detection and the fitted blendshape model. The sparse set of 2D facial landmarks are used to initialize the 3D facial mesh. Textures are extracted from the images and used to track the performance densely in a per-pixel fashion.*

Pose estimation

Following Snavely et al. (2010) we use a pinhole camera model with radial distortion. Assuming the pixels are square and that the centre of projection is coincident with the image centre, the projection operation $\prod$ depends on 10 parameters: the 3D orientation $\mathbf{r}$ (3×1 vector), the translation $\mathbf{t}$ (3×1 vector), the focal length f (scalar), and the distortion parameter $\mathbf{k}$ (3×1 vector).

We assume the same distortion and focal length for the entire video, and initialize the focal length to be the pixel width of the video and distortion to zero. First, we apply a direct linear transform (Chen et al. 1994) to estimate the initial rotation and translation then optimize them via the LevenbergMarquardt method with a robust loss function (Moré 1978). The derivative of the pose parameter to the alignment objective function can be derived analytically as in Sec. 4.1.1 or computed via forward accumulation automatic differentiation (Rall 1981).

Expression estimation

In the pose estimation stage, we used a generic face model for initialization, but to get more accurate results we need to adjust the model according to

the expression and identity. We use the FaceWarehouse dataset (Cao et al. 2014b), which contains the performances of 150 people with 47 different expressions. Since we are only tracking facial expressions, we select only the frontal facial vertices because the nose and head shape are not included in the detected landmarks.

We flatten the 3D vertices and arrange them into a 3 mode data tensor. We compress the original tensor representing 30k vertices $\times$ 150 identities $\times$ 47 expressions into a 4k vertices $\times$ 50 identities $\times$ 25 expression coefficients core using higher order singular value decomposition (Kolda and Sun 2008). Any facial mesh in the dataset can be approximated by the product of its core $B_{exp} = C \times U_{id}$ or $B_{id} = C \times U_{exp}$, where U_{id} and U_{exp} are the identity and expression orthonormal matrices respectively; B_{exp} is a person with different facial expressions, B_{id} is the same expression performed by different individuals.

For efficiency we first determine the identity with the compressed core and prevent over-fitting with an early stopping strategy. To generate plausible results we need to solve for the uncompressed expression coefficients with early stopping and box constrain them to lie within a valid range, which in the case of FaceWarehouse is between 0 and 1. We do not optimize identity and camera coefficients for individual frames. They are only optimized jointly after expression coefficients have been estimated.

We group the camera parameters into a vector $\theta = [\mathbf{r}, \mathbf{t}, f]$. We generate a person specific facial mesh B_{id} with this persons identity coefficient I, which results in the same individual performing the 47 defined expressions. The projection operator is defined as $\prod([x, y, z]^T) = \mathbf{R}[x, y, z]^T + \mathbf{t}$, where $\mathbf{R}$ is the 3×3 rotation matrix constructed $\mathbf{r}$ and the radial distortion function

$\mathbb{D}$ is defined as:

$$\begin{aligned}
\mathbb{D}(X', k) &= f \times X'(1 + k_1 r^2 + k_2 r^4) \\
\mathbb{D}(Y', k) &= f \times Y'(1 + k_1 r^2 + k_2 r^4) \\
r^2 = X'^{,2} + Y'^2, X' &= X/Z, Y' = Y/Z \\
[X, Y, Z]^T &= \prod([x, y, z]^T)
\end{aligned} \tag{4.15}$$

We minimize the squared distance between the 2D landmarks L after applying radial distortion while fixing the identity coefficient and pose parameters $\mathbb{D}$:

$$\min_E \frac{1}{2}||L - \mathbb{D}(\prod(B_{id} \cdot E, \theta), k)|| \tag{4.16}$$

To solve this problem efficiently, we apply the reverse distortion to L, then rotate and translate the vertices. By denoting the projected coordinates by p, the derivative of E can be expressed efficiently as:

$$(L - f \cdot p)(f \cdot \frac{B^{(i)}_{id(0,1)} + B^{(i)}_{id(2)} \cdot p}{Z}), \tag{4.17}$$

where we use the LevenbergMarquardt method for initialization and perform line search (Li and Fukushima 2001) with a upper and lower box constraints to force E to lie within the valid range.

Identity adaptation

Since we cannot apply a generic B_{id} to different individuals with differing facial geometry, we solve for the subject's identity in a similar fashion to the expression coefficient. With the estimated expression coefficients from the last step, we generate facial meshes of different individuals performing the estimated expressions. Unlike expression coefficient estimation, we need to solve identity coefficient jointly across I frames with different poses and ex-

pressions. We denote the nth facial mesh by B^n_{exp} and minimize the distance:

$$\min_I \sum_n \frac{1}{2}||L^n - \mathbb{D}(\prod(B^n_{exp} \cdot I, \theta), k)||, \tag{4.18}$$

while fixing all other parameters. Here it is important to exclude inaccurate single frames from being considered otherwise they lead to erroneous identity.

Camera estimation

Some videos may be captured with camera distortions. In order to reconstruct the 3D facial geometry as accurately as possible, we undistort the video by estimating its focal length and distortion parameters. All of the following dense tracking is performed in undistorted camera space. To avoid local minima caused by over-fitting the distortion parameters, we solve for focal length analytically using:

$$f = \frac{\sum_n L^n}{\sum_n \mathbb{D}(\prod(B^n_{exp} \cdot I, \theta), k)} \tag{4.19}$$

then use nonlinear optimization to solve for radial distortion. We find the camera parameters by jointly minimizing the difference between the selected 2D landmarks L and their corresponding projected vertices:

$$\min_k \sum_n \frac{1}{2}||L^n - \mathbb{D}(\prod(B^n_{exp} \cdot I, \theta), k)|| \tag{4.20}$$

Average texture estimation

In order to estimate an average texture, we extract per pixel colour information from the video frames. We use the texture coordinates provided in FaceWarehouse to normalize the facial texture onto a flattened 2D map. By performing visibility tests we filter out invisible pixels. Since the eyeball and inside of the mouth are not modelled by facial landmarks or FaceWarehouse,

we consider their texture separately. Although varying expressions, pose, and lighting conditions lead to texture variation across different frames, we use their summed average as a low rank approximation.

Alternatively, we could use the median pixel values as it leads to sharper texture, but at the coarse reconstruction we choose not to because computing the median requires all the images to be available whereas the average can be computed on-the-fly without additional memory costs. Moreover, while the detected landmarks are not entirely accurate, robustness is more important than accuracy. Instead, we selectively compute the median of high quality frames from dense reconstruction to generate better texture in the next stage.

The idea of tracking the facial landmarks by minimizing the difference between synthesized view and the real image is similar to that used in active appearance models (AAM) (Cootes et al. 2001). The texture variance can be modelled and approximated by principle component analysis, and expression pose specific texture can be used for better performance. Experimental results show that high rank approximation leads to unstable results because of the landmark detection in-the-wild issues. Moreover, AAM typically has to be trained on manually labelled images that are very accurate. Although it is able to fit the test image with better texture similarity, it is not suitable for robust automated landmark detection. A comparison of our method with traditional AAM method and examples of failed detections are shown in Fig. 4.11.

Up to this point, we have been optimizing the 3D coordinates of the facial mesh and the camera parameters. Due to the limited expressiveness of the facial dataset, which only contains 150 persons, the fitted facial mesh might not exactly fit the detected landmarks. To increase the expressiveness of the reconstructed model and add more person specific details, we use the method in (Igarashi et al. 2005) to deform the facial mesh reconstructed for each frame. We first assign the depth of the 2D landmarks to that of their corresponding 3D vertices, then unproject them into 3D space. Finally, we use the unprojected 3D coordinates as anchor points to deform the facial mesh of every frame.

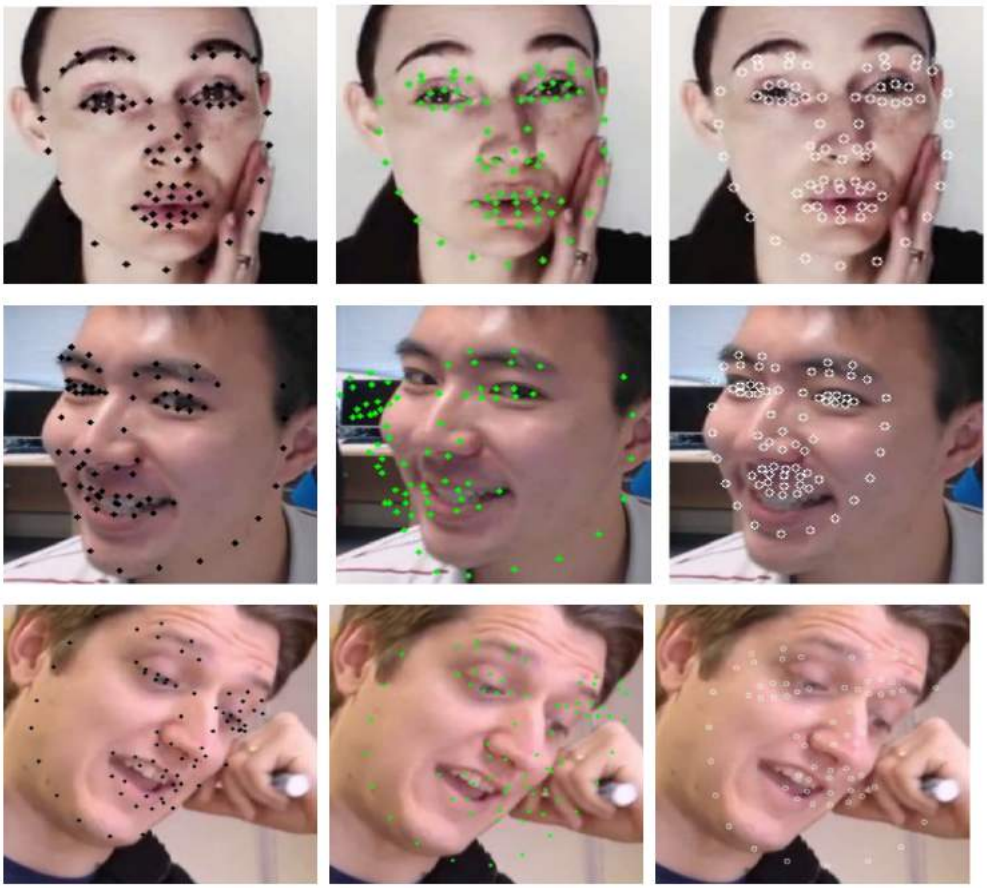
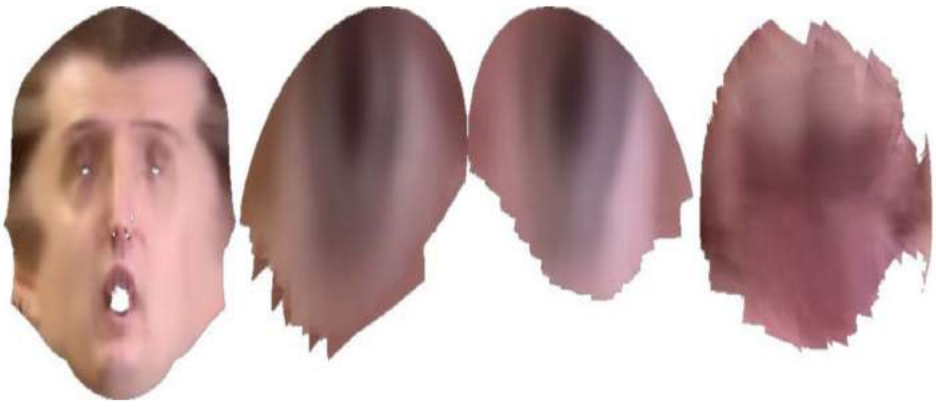


Figure 4.11: *Qualitative landmark detection results of our method compared to AAM and ERT. Our results are shown on the left, AAM in the middle and ERT on the right. It can be observed that our method produces more accurate results in challenging scenarios.*

Since the deformed facial mesh may not be represented by the original data, we need to add them into the person specific facial meshes B_{exp} and keep the original expression coefficients. Given an expression coefficient E we could reconstruct its corresponding facial mesh $F = B_{exp}E$. Thus the new deformed mesh base should be computed via $F_d = B_d E_d$. We flatten the deformed and original facial meshes using B_{exp}, then concatenate them together as $Bc = [B; B_d]^T$. We concatenate coefficients of the 47 expressions in FaceWarehouse and the recovered expressions from the video frames as $E_c = [E; E_d]^T$. The new deformed facial mesh base is computed from $B_d = E_c^1 B_c$.

(a) *Sparse texture*



(b) *Dense texture*

Figure 4.12: *Refined texture on the bottom after dense tracking are sharper compared to the original texture on the top. It can be observed that the texture recomputed from densely tracked results are sharper and more accurate than the ones extracted from sparse facial landmark alignment.*

We simply compute for each pixel the average colour value and run the

k-means algorithm (Hartigan and Wong 1979) on the extracted eyeball and mouth interior textures, saving a few representative k-means centers for fitting different expressions and eye movements. An example of the reconstructed average face texture is shown in Fig. 4.12a.

4.2.3 Dense reconstruction to refine landmarks

Face tracking flow

In the previous step we reconstructed an average face model with a set of coarse facial landmarks. To deliver convincing results we need to track and reconstruct all of the vertices even in weakly textured areas. To robustly capture the 3D facial performance in each frame, we formulate the problem in terms of 3D optical flow and solve for dense correspondence between the 3D model and each video frame, optimally deforming the reference mesh to fit the seen image. We use the rendered average shape as initialization and treat it as the previous frame; we use the real image as the current frame to densely compute the displacement of all vertices. Assuming the pixel intensity does not change by the displacement, which is written as (Horn and Schunck 1981):

$$I(x, y) = C(x + u, y + v), \tag{4.21}$$

where I denotes the intensity value of the rendered image, C the real image, and x and y denote pixel coordinates. In addition, the gradient value of each pixel should also not change due to displacement because not only the pixel intensity but also the texture stay the same, which can be expressed as:

$$\Delta I(x, y) = \Delta C(x + u, y + v) \tag{4.22}$$

Finally, the smoothness constraint dictates that pixels should stay in the same spatial arrangement to their original neighbours to avoid the aperture problem, especially since many facial areas are weakly textured, i.e., have no

strong gradient. We search for $f = (u, v)^T$ that satisfies the pixel intensity, gradient, and smoothness constraints.

By denoting each projected vertex of the face mesh by $p = \mathbb{D}(\prod(B_{id}^n \cdot E, \theta), k)$, we formulate the energy as:

$$E_{flow}(f) = \sum_v ||I(p+f) - C(p)||^2 = \alpha(||\Delta f||2) + \beta(||\alpha f||^2) \quad (4.23)$$

Here $||\Delta f||^2$ is a smoothness term and $\beta(||\alpha f||^2)$ is a piecewise smooth term. As this is a highly nonlinear problem we adopt the numerical approximation in (Brox et al. 2004) and take a multi-scale approach to achieve robustness. We do not use the additional matching term in (Brox and Malik 2011), because if though we have the match from the landmarks to the vertices, we cannot measure the quality of the landmarks, as well as the matches.

Robust tracking

Standard optical flow suffers from drift, occlusion, and varying visibility because of lack of explicit modelling. Since we already have a rough prior of the face from the coarse reconstruction step, we use it to correct and regularize the estimated optical flow.

We test the visibility of each vertex by comparing its transformed value to its rendered depth value. If it is larger than a threshold then it is considered to be invisible and not used to solve for pose and expression coefficient. To detect partially occluded areas we compute both the forward flow (rendered to real image f_f) and backward flow (real image to rendered f_b), and compute the difference for projections of each vertices:

$$\sum_p ||f_f(p) + f_b(p + f_f(p))||^2 \quad (4.24)$$

We use the GPU to compute the flow field whereas the expression coef-

ficient and pose are computed on the CPU. Solving them for all vertices can be expensive when there is expression and pose variation, so to reduce the computational cost, we also check the norm of $f_f(p)$ to filter out pixels with negligible displacement.

Because of the piecewise smoothness constraint, we consider vertices with large forward and backward flow differences to be occluded and exclude them from the solution process. We first find the rotation and translation, then the expression coefficients after putative flow fields have been identified. The solution process is similar to that used in the previous section with the exception that we update each individual vertex at the end of the iterations to fit the real image as closely as possible. To exploit temporal and spatial coherence, we use the average of a frames neighbouring frames to initialize its pose and expression, then update them using coordinate descent. If desired, we reconstruct the average face model and texture from the densely tracked results and use the new model and texture to perform robust tracking again. An example of updated reconstructed average texture is shown in Fig. 4.12, which is sharper and more accurate than the coarsely reconstructed texture. Filtered vertices and the tracked mesh are shown in Fig. 4.13, where putative vertices are colour coded and filtered out vertices are hidden. Note that the colour of the actress hand is very close to that of her face, so it is hard to mask out by colour difference thresholding without piecewise smoothness regularization.

Texture update

Finally, after robust dense tracking results and the validity of each vertex have been determined, each valid vertex can be optionally optimized individually to recover further details. This is done in a coordinate descent manner with respect to the pose parameters. Updating all vertices with a standard non-linear optimization routine might be inefficient because of the computational cost of inverting or approximating a large second order Hessian matrix, which is sparse in this case because the points do not have influence on each other.

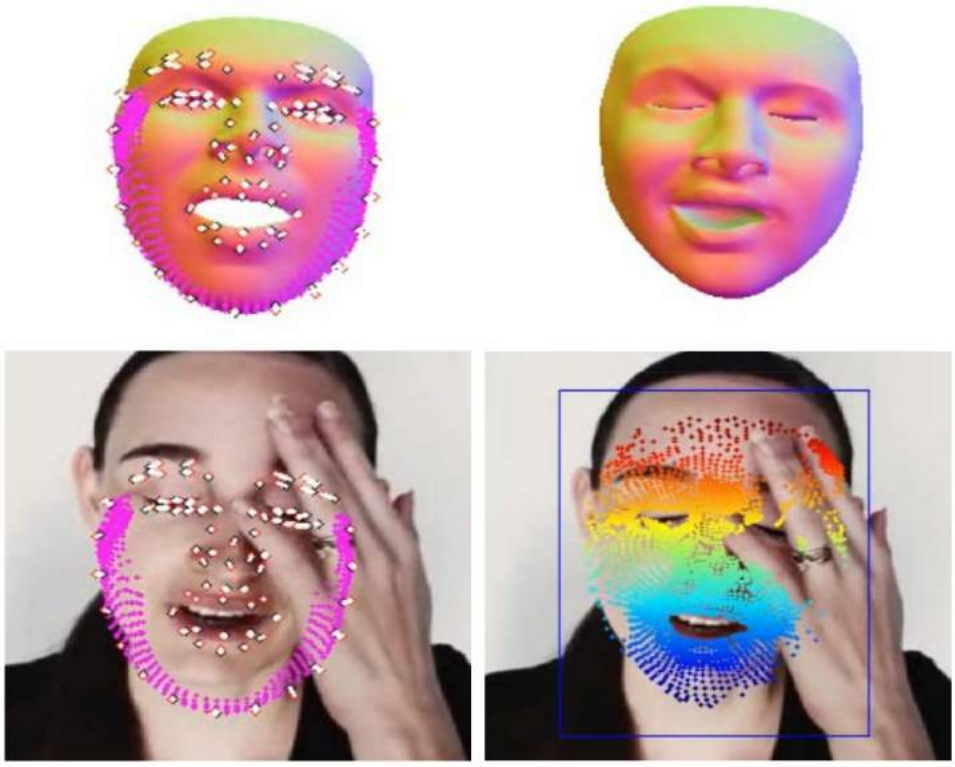


Figure 4.13: *Example of robust facial performance tracking results with the presence of partial occlusion. Only valid vertices of the mesh are drawn with colour on the bottom right figure, where the parts occluded by the hand are masked out. It can be observed the the densely tracked results are more accurate to the ones from sparse landmarks on the left.*

Thus, instead, we use the Schur complement trick (Agarwal et al. 2010) to reduce the computational cost. The whole pipeline of our method is summarized in Algorithm 2. Convergence is determined by the norm of the optical flow displacement. This criterion indicates whether further vertex adjustment is possible or necessary to minimize the difference between the observed image and synthesized result.

4.2.4 Experiments

Our proposed method aims to deliver smooth facial performances and landmark tracking in uncontrolled in-the-wild videos. Although recently a new dataset has been introduced designed for facial landmark tracking in the wild (Shen et al. 2015), it is not adequate for this work since we aim to deliver smooth tracking results rather than just locating landmark positions. In addi-

Algorithm 2: Automatic dense facial capture

```
input : Video
CCA-GSDM landmark detection;
Solve Pose, Expression, Identity, Focal and Distortion on landmarks;
while not converged do
    while norm(flow) >threshold do
        Determine vertex validity via depth, Eq. 4.24 and flow
          displacement norm;
        Solve Pose and Expression on optical flow;
        if Inner max iteration reached then
            break;
    Update camera, vertex and texture;
    if Outer max iteration reached then
        break;
output: Facial meshes, poses, expressions
```

tion, we also concentrate on capturing detail to reconstruct realistic expressions. Comparison of the expression norm between the coarse landmarks and dense tracking is shown in Fig. 4.14.

In order to evaluate the performance of our robust method, AAM [3, 22], and an in-the-wild regressorbased method (Ren et al. 2014; Kazemi and Sullivan 2014) working as *fully automated* methods, we collected 50 online videos with frame counts ranging from 150 to 897 and manually labeled them. Their resolution is 640×360. There are a wide range of different poses and expressions in these videos, and heavy partial occlusion as well. Being *fully automated* means that given any in-the-wild video no more additional effort is required to tune the model. We manually label landmarks for a quarter of the frames sampled uniformly throughout the entire video to train a person specific AAM model then use the trained model to track the landmarks. Note that doing so disqualifies the AAM approach as a *fully automated* method. Next we manually correct the tracked result to generate a smooth and visually plausible landmark sequence. We treat such sequences as ground truth and test each methods accuracy against it. We also use these manually labelled landmarks to build corresponding coarse facial models and texture in a similar way to the approach used in Section 3. The result is shown in Table 1. Each numeric column represents the error between the ground truth and

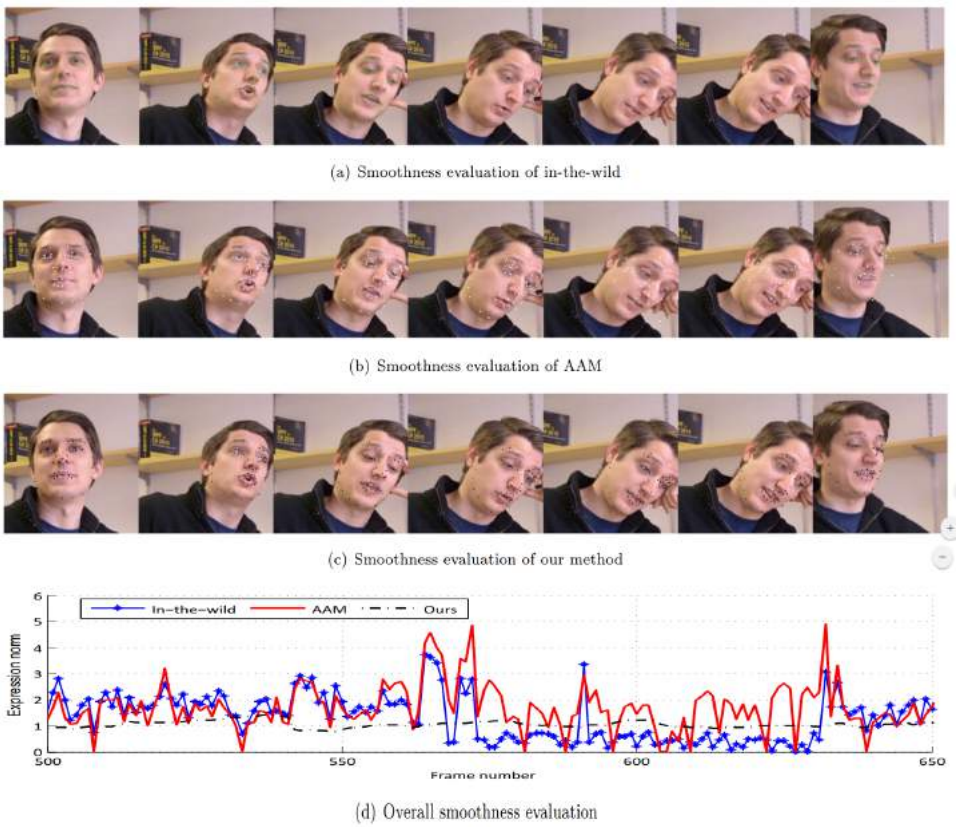


Figure 4.14: *Example qualitative tracking results and the expression coefficient smoothness comparison. The facial expression coefficient difference for consecutive frames are also shown in the graph, which shows that our method is more stable.*

the methods output. Following standard practice (Cao et al. 2014c; Ren et al. 2014; Belhumeur et al. 2013), we use the inter-pupillary distance normalized landmark error. Mesh reconstruction error is measured by the average L2 distance between the reconstructed meshes. Texture error is measured by the average of per-pixel colour difference between the reconstructed textures.

We mainly compare our method to appearance based methods (Cootes et al. 2001; Donner et al. 2006) and in-the-wild methods (Ren et al. 2014; Kazemi and Sullivan 2014) because they are appropriate for in-the-wild video and have similar aims to minimize texture discrepancy between synthetic views and real images. We have also built a CUDA-based face tracking application using our method; it can achieve realtime tracking. Benefiting from the CUDA speed-up, our methods runs at more than 30fps on the tested video with a resolution of 640×360. The dense points (there are 5760 of them) are

Method	Mesh	Texture	Landmark
Ours 1 iteration	**13.3**	**29.2**	**4.4**
Ours 2 iteration	**10.3**	**25.4**	**3.2**
Kazemi and Sullivan (2014)	33.2	31.7	9.7
Ren et al. (2014)	37.8	24.9	7.4
Donner et al. (2006)	23.3	47.2	15.2
Cootes et al. (1998)	24.3	56.1	24.4
Low	41.3	136.8	35.4
High	54.3	186.5	32.2

Table 4.1: *Performance evaluation of the robust tracking on the whole dataset, each column shows the error of the method and task respectively. Our method achieved the best results in the first two iterations.*

from the frontal face of a standard blendshape mesh.

For completeness we also used the detected landmarks obtained from in-the-wild methods to train the AAM models, then used these to detect landmarks in videos. Doing so qualifies them as fully automated methods again. Due to the inconsistent results produced by in-the-wild landmark detectors, we use both high and low rank texture approximation thresholds when training the AAM. Note that although Donner et al. (2006) propose use of regression relevant information which may be discarded by purely generative PCA based models, they also use an approximate texture variance model. Models trained with low rank variance are essentially the same as our approach of just taking the average of all images. While low rank AAM can accurately track the pose of the face most of the time when there is no large rotation, it fails to track facial point movements such as closing and opening of eyes, and talking, because the low rank model limits its expressiveness. High rank AAM, on the other hand, can track facial point movements but produces unstable results due to the instability of the training data provided by the in-the-wild method. Experimental results of training AAM with landmarks detected by the method in (Kazemi and Sullivan 2014) are shown in the *Low* and *High* columns of Table 4.1.

We also considered separately a challenging subset of the videos, in which there is more partial occlusion, large head rotation or exaggerated facial expression. The performance of each method is given in Table 4.2. A

Method	Mesh	Texture	Landmark
Ours 1 iteration	**41.7**	**59.1**	**7.2**
Ours 2 iteration	**15.1**	**35.2**	**4.1**
Kazemi and Sullivan (2014)	92.3	95.4	19.2
Ren et al. (2014)	88.1	114.3	11.4
Donner et al. (2006)	97.3	142.7	21.9
Cootes et al. (1998)	87.9	136.2	21.3
Low	114.3	146.8	25.3
High	134.3	186.2	33.4

Table 4.2: *Performance evaluation of the robust tracking on the challenging dataset, each column shows the error of the method and task respectively. Our method achieved the best results in the first two iterations.*

comparison of our method to AAM and the in-the-wild method is shown in Fig. 4.14, where the x axis is the frame count and the y axis is the norm of the expression coefficient. Compared to facial performance tracking with only coarse and inaccurate landmarks, our method is very stable and has a lower error rate than the other two methods. Further landmark tracking results are shown in Fig. 4.15. Additional results and potential applications are shown in the Electronic Supplementary Material.

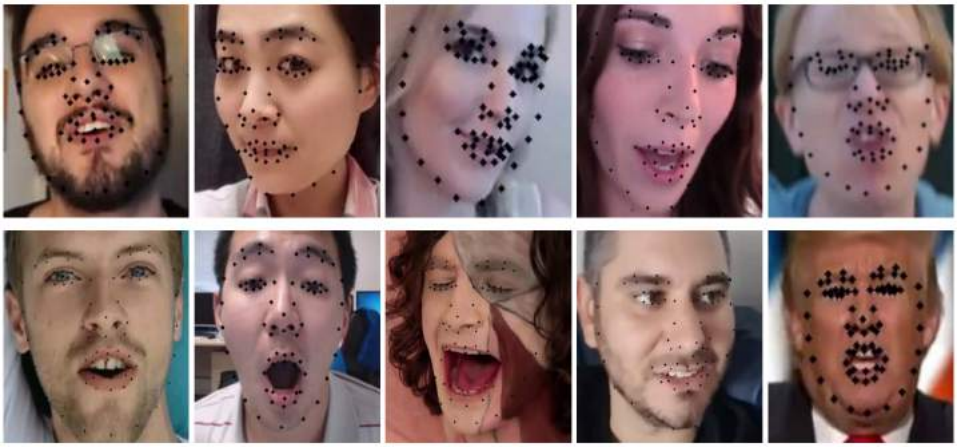


Figure 4.15: *Additional qualitative robust tracking results from test videos. Our method works on a variety of videos and deliver accurate results regardless of the video quality.*

4.2.5 Summary

In this section we have proposed a novel fully automated method for robust facial landmark detection and tracking across poses and expressions for in-

the-wild monocular videos. In our work, shape-face canonical correlation analysis is combined with a global supervised descent method to achieve robust coarse 2D facial landmark detection across poses.

We perform coarse-to-dense 3D facial expression reconstruction with a 3D facial prior to boost tracked landmarks. We have evaluated its performance with respect to existing landmark detection methods and empirically compared the tracked results to those of conventional approaches. Compared to conventional tracking methods that are able to capture subtle facial movement details, our method is fully automated, just as expressive and robust in noisy situations.

Compared to other robust in-the-wild methods, our method delivers smooth tracking results and is able to capture small facial movements even for weakly textured areas. Moreover, we can accurately compute the possibility of a facial area being occluded in a particular frame, allowing us to avoid erroneous results. The 3D facial geometry and performance reconstructed and captured by our method are not only accurate and visually convincing, but we can also extract 2D landmarks from the mesh and use them in other methods that depend on 2D facial landmarks, such as facial editing, registration, and recognition.

Currently we are only using the average texture model for all poses and expressions. To further improve the expressiveness, we could adopt a similar approach to that taken for active appearance models, where after we have robustly built an average face model, texture variance caused by different lighting conditions, pose and expression variation could also be modelled to improve the expressiveness and accuracy of the tracking results.

Chapter 5

3D Facial Geometry Reconstruction

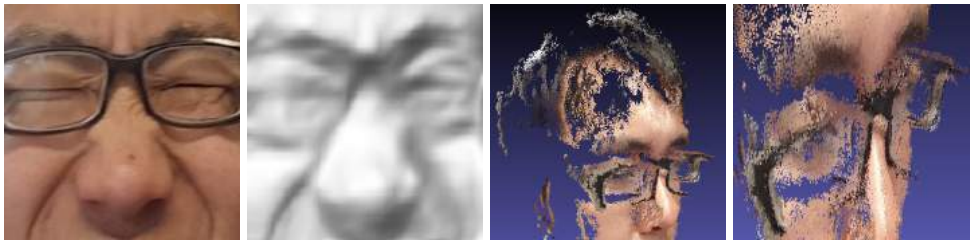
With the increasing amount of videos recorded using 2D mobile cameras, the technique for recovering the 3D dynamic facial models from these monocular videos has become a necessity for many image and video editing applications. While methods based on parametric 3D facial models can reconstruct the 3D shape in dynamic environment, large structural changes are ignored. Structure-from-motion methods can reconstruct these changes but assume the object to be static. To address this problem, in this chapter we present a novel method for realtime dynamic 3D facial tracking and reconstruction from videos captured in uncontrolled environments. Our method can track the deforming facial geometry and reconstruct external objects that protrude from the face such as glasses and hair. It also allows users to move around, perform facial expressions freely without degrading the reconstruction quality.

3D facial modelling is an essential technique for animation production in featured films and video games. Dedicated hardware such as depth sensors, laser scanners and camera arrays have been developed to acquire depth information for 3D model creation. However these can only be operated by trained professionals. In recent years, the wide spread availability of 2D RGB mobile cameras has sparked interest in 3D facial reconstruction from 2D in-

put, due to the increased interest of casual untrained users in applications such as image, video editing (Thies et al. 2016; Fried et al. 2016), virtual makeup(Scherbaum et al. 2011) and facial model creation (Cao et al. 2014b).

5.1 Limitation of existing methods

Existing works based on parametric 3D facial model and shape-from-shading (Suwajanakorn et al. 2014; Cao et al. 2015; Garrido et al. 2013) are able to reconstruct minuscule detail while allowing the user to move around freely in the monocular setting. However, these methods cannot deal with structures such as hair and glasses (Fig. 5.1b). SFM methods (Hartley and Zisserman 2003; Furukawa and Ponce 2010; Ichim et al. 2015), which estimate 3D structures from 2D images with different viewing angles, are able to handle these large variations. Nevertheless, the user is required to remain still while images from different angles are being taken. It involves separate capture and off-line processing phases, which is suboptimal and tedious because they require careful planning and possibly numerous trials. Moreover, feature point detection and matching on facial areas such as the cheek and forehead are more likely to fail due to the lack of highly distinctive texture pattern. Furthermore, without constraints such as controllable lighting, camera focus and limited motion, extra post-processing effort, like manual landmark adjustment, user specific model crafting and texture creation are inevitable even for state-of-the-art techniques (Garrido et al. 2013, 2016; Thies et al. 2016). Recently a method was proposed in (Cao et al. 2016), which is able to reconstruct hair but requires user input and interaction to specify 2D hair boundaries of images taken from different angles.



(a) **(b)** **(c)** **(d)**

Figure 5.1: *Shape from shading methods (Suwajanakorn et al. 2014; Barron and Malik 2015) can recover minuscule details such as wrinkles, shown in 5.1a and 5.1b, however they cannot reconstruct larger geometry variation such as hair and glasses, which SFM methods (Furukawa and Ponce 2010; Galliani et al. 2015) can handle but fail to produce complete or smooth surface shown in 5.1c and 5.1d.*

To address this challenge, we propose a novel realtime method that aims at automatically tracking the 3D facial performance and reconstructing the 3D geometry from monocular videos in uncontrolled environment. In order to reconstruct a dynamically deforming object, it is essential to define a rigid reference for the object. Although defining a canonical rigid reference for general object is not straightforward, facial deformation can be represented as facial expression variation. Therefore, we reconstruct the dynamic facial geometry by undoing the deformation caused by different expressions, which is made possible by a robust 3D facial tracking method. The flowchart of the proposed method is illustrated in Fig. 5.2.

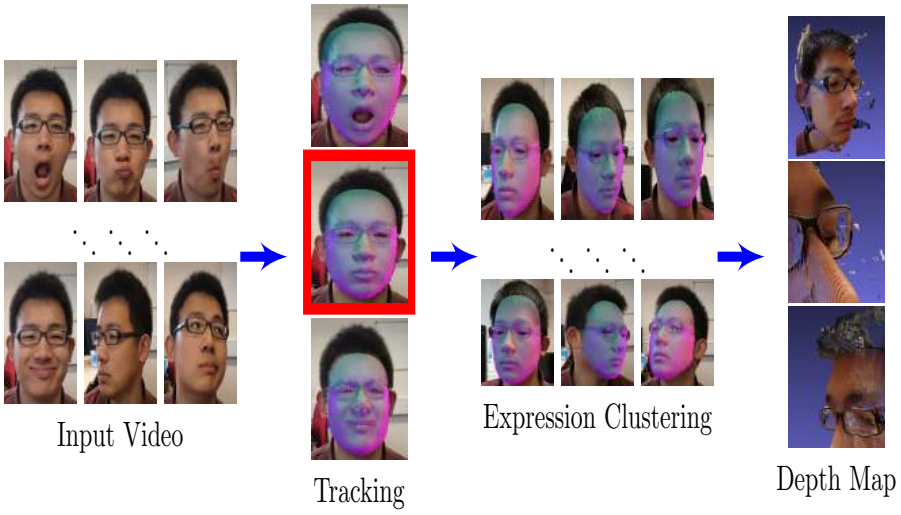


Figure 5.2: *Given a video, a parametric 3D model (Cao et al. 2014b) was fitted to noisy 2D landmarks produced from the off-the-shelf 2D landmark detector (Kazemi and Sullivan 2014). The fitted 3D model is used to refine the 3D position computed from the 2D landmarks via robust photometric tracking. The tracked 3D mesh are superimposed on the image. After clustering, video frames of similar facial expressions as seen at different viewing angles are used to compute a complete and smooth dense depth map. The glasses and hair are well reconstructed.*

Existing 3D facial **tracking** and **reconstruction** works could be categorized as *depth* based and 2D *image* based. Although depth based methods are inherently less likely to suffer from depth ambiguity, they require special depth sensors, and therefore cannot process the vast majority of 2D recordings. Moreover, consumer grade depth sensors often fail to capture high frequency shape detail. Binocular and stereo vision systems are able to overcome the resolution limits but require careful synchronization.

Ever since the release of consumer grade device such as Kinect, various methods that operate on noisy *depth* input have been proposed. A depth based method was introduced, which uses a parametric 3D facial model to robustly deal with the noisy depth input in Weise et al. (2011). Recently a state-of-the-art depth based tracking method with parametric 3D facial geometry and lighting model has been proposed for realtime facial expression transfer and re-enactment in Thies et al. (2015). Due to the limited depth sensor resolution, RGB color input is used to supplement extra information to

refine the tracking. An adaptive scheme was proposed to capture more detail with point-to-point deformation on top of blendshapes in Li et al. (2013). To explicitly deal with outliers caused by occlusions, a method was proposed to segment the face and complete the occluded parts based on the blendshape in Hsieh et al. (2015), which was later extended to RGB input in Saito et al. (2016). Binocular stereo system, on the other hand, can provide higher resolution depth map and work in outdoor environments directly under sunlight, but are more prone to suffer from lighting variation. A robust method was introduced for a lightweight binocular system under uncontrolled lighting in Valgaerts et al. (2012). Generally, for most of these state-of-the-art methods, a parametric 3D facial model is first fitted for the tracking target, which is later used for 3D tracking. These methods are quite stable as the combination of depth information the 3D facial model can effectively eliminate outliers and uncertainty.

2D *image* based methods are capable of processing existing footage without depth information, and are also more flexible in terms of hardware setup requirement. However, due to the lack of depth they are more likely to suffer from depth ambiguity and lighting variation. In Cao et al and Garido et al Garrido et al. (2013); Cao et al. (2013) works personalized 3D facial models are first crafted for the tracking target semi-manually, which is later used to track the performance. Later a dynamic method was proposed to automatically track and generate personalized facial blendshape in realtime in Cao et al. (2014a). Built on top of the robust tracking, more details were added to the person specific blendshape based on image input and user interaction in Cao et al. (2015, 2016).

The creation of photo-realistic person specific facial model can be useful for many application as seen in Thies et al. (2016); Liu et al. (2016), where the tracked facial performance was used to transfer the expression of source actor to the target. In order to create person specific model from a monocular rig, a method was proposed in Ichim et al. (2015), which produces facial mesh via multi-view stereo vision pipeline. To allow the geometry deformation and variations go beyond the blendshape and minuscule details, a method was

proposed in Wu et al. (2016), which physically models the anatomical structure of the face and deform the person specific model to match the monocular video input.

In summary, to the best of our knowledge, none of the reviewed works directly address the challenge of delivering a mobile-user-friendly tool that is capable of dynamically reconstructing full-scale 3D facial geometry in uncontrolled environments. To satisfy this need, we firstly propose a robust realtime 3D facial tracking method in Section 5.2, which obtains blendshape coefficient that is used to categorize facial deformation. After that, a novel depth reconstruction method is introduced in Section 5.3, where depth map is estimated for each individual expression. The performance of our method is examined in Section 5.4. We evaluate the quality of 3D tracking and depth reconstruction by comparing our method against popular methods Kazemi and Sullivan (2014); Cao et al. (2014a); Furukawa and Ponce (2010); Galliani et al. (2015).

5.2 Robust Tracking

5.2.1 Parametric Model Fitting

To initialize, we first apply an off-the-shelf face detector (Viola and Jones 2004) to obtain the bounding boxes. A 2D facial landmark detector (Kazemi and Sullivan 2014) is trained on the dataset in Cao et al. (2014a), which has 73 landmarks that include essential facial features on the eyebrows, nose, mouth and the face contours, which are necessary in determining the neutral face shape and expression variation. In our experiments the landmark detector in Kazemi and Sullivan (2014) achieved the best trade-off between efficiency and accuracy, but for applications where real-time is not a priority the landmark detector could be swapped by more robust ones such as Zhang et al. (2016, 2017). To reduce redundancy only representative landmarks are chosen as described in Liu et al. (2017) as well. The landmarks in frame i

is denoted as S_i, and the 3D parametric model from Cao et al. (2014b) is represented as:

$$T \times_2 U_{id}^T \times_3 U_{exp}^T = C, \tag{5.1}$$

where T is the data tensor and C is the core tensor. U_{id} and U_{exp} are orthonormal transform matrices, which contain the left singular vectors of the 2nd mode (identity) space and 3rd mode (expression) space respectively. In our setup we found that choosing 50 knobs for identity and 25 knobs for expression provides satisfactory approximation results.

The perspective projection operator is denoted as Π and the camera matrix is expressed as: $A = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$. The intrinsic camera parameters are solved via the coordinate descent approach by iteratively fixing either the intrinsic camera parameters or the remaining parameters and solve for the others. The indices of inner facial landmarks such as nose, eyes, eyebrows and mouth are fixed, and the indices of face contours are updated in each iteration. The face contours are computed by uniformly sampling from the convex hull of projected facial contour vertices. The 3D to 2D alignment problem is solved by minimizing

$$\min_{I,E_i,R_i,t_i} \sum_i \left\| \Pi_A(I, E_i, R_i, t_i) - S_i \right\|_\epsilon, \tag{5.2}$$

where I, E_i, R_i and t_i denotes the identity coefficient for all frames, expression coefficients and the 3D rodrigues rotation and translation vector for frame i respectively. To minimize the effect of outliers in landmark detection, the robust Huber loss is applied, where ϵ controls the tolerance for outliers.

$$\left\| \delta \right\|_\epsilon = \begin{cases} \frac{1}{2}\delta^2 & \text{for} |\delta| \leq \epsilon, \\ \epsilon \left| \delta \right| - \frac{1}{2}\epsilon^2 & \text{otherwise.} \end{cases} \tag{5.3}$$

The projection function is nonlinear but differentiable. Firstly the rotation and translation are solved via direct linear transform. Then all parameters are solved jointly via the Levenberg Marquardt algorithm Moré (1978). Empirical experiments show that trust region optimization method converges to more natural expression, identity coefficient and smaller error, than line search and coordinate descent methods. Since the identity coefficient is the only parameter that affects all frames, the zero pattern in the normal equations is exploited to reduce the computational cost Davis (2006). To keep the blendshape within valid range a box constraint is simulated which clamps the expression and identity coefficient parameter within the column-wise minimum l and maximum u of U_{id}^T and U_{exp}^T. The box constraint is achieved by variable transformation as:

$$f(x) = \frac{u+l}{2} + \frac{u-l}{2} \cdot \tanh((x - \frac{u+l}{2})/\frac{u-l}{2}), \tag{5.4}$$

and the corresponding transformed partial derivative is

$$f'(x) = x - x \cdot \tanh((x - \frac{u+l}{2})/\frac{u-l}{2})^2. \tag{5.5}$$

5.2.2 Photometric Tracking

To robustly track the object in new incoming frames, the photometric difference between the rendering and image is minimized, which greatly benefits the tracking quality because of automatic occlusion handling back faces culling. Given the parametric model computed from a few landmarks and images, realistic rendering is synthesized for previously unseen angles.

Due to the noisy environment and complex lighting in real world situations, we propose to simply use the median of extracted surface maps as a robust approximation of the face texture, which is updated during tracking. Empirically the experimental results show that such a low cost and straightforward approximation achieves similar performance to existing works that explicitly estimate the illumination and albedo of the face. The smoothness

term is applied on a small window of 10 frames because longer sequences might not necessarily improve the accuracy, and slow down the computation. As a result, instant feedback of tracked 3D performance is provided since it is not essential for the 3D reconstruction.

Given the ith frame, we define the rendering function as Φ and the target energy as:

$$\min_{P} \sum_{i} \|\Phi(\Pi_A(\mathbf{P_i})) - F_i\|_\epsilon + \beta \cdot \Theta(\mathbf{P}), \tag{5.6}$$

where $\mathbf{P}$ denotes the set of parameters E, R and t that are used to synthesize a virtual view given the texture map. A L2 smoothness term $\Theta(X)$ controlled by β is used on the expression and pose parameters is to exploit the temporal coherence, which is defined as

$$\Theta(X) = \|XO\|, \tag{5.7}$$

where $O \in \mathbb{R}^{n \times n}$ is a symmetrical matrix defined by

$$O = \begin{pmatrix} -1 & 1 & & & & & \\ 1 & -2 & 1 & & & & \\ & \ddots & \ddots & \ddots & & & \\ & & 1 & -2 & 1 & & \\ & & & & 1 & -1 \end{pmatrix}. \tag{5.8}$$

The eigen-decomposition of $\beta \cdot \Theta(\mathbf{P})$ is actually the n-by-n type-2 discrete cosine transform (DCT) and inverse DCT (IDCT) matrices and can be directly solved Garcia (2010).

Both the rendering and photometric evaluation function carry high computational cost. The search radius of E, R and t can be clamped to reduce the computational cost, which is calculated with respect to β as in Equation. (5.4, 5.5). To make intensity difference term $\rho = \|\Phi(\Pi_A(\mathbf{P})) - F_i\|_\epsilon$ differentiable, it is linearised via Taylor expansions approximation yielding the

following equation:

$$\rho = \Phi(\Pi_A(\mathbf{P})) - \nabla F(x+u) \cdot (p-u) - F(x+p), \tag{5.9}$$

where u is image coordinate difference in F and p is the projected coordinate, ∇F is computed from 3×3 Sobel kernel convolution. The computational cost of evaluating the simulated Hessian matrix in trust-region methods at per-pixel level becomes a bottle neck. Hence we switch to line search method Bonnans et al. (2006) with simulated Hessian matrix computed from previous gradient directions.

Even with reduced search radius, evaluating per-pixel is expensive when the input resolution is high. We take advantage of the high parallel capacity of GPU to achieve lower latency. The face reconstructed from the core C only contains points on the mesh, hence we render per-vertex smoothed coordinates and colour of the mesh with as a texture, then use CUDA/OpenGL interoperability to directly read from GPU memory and evaluate the cost function and derivative on GPU. It is only necessary to update the rendering in outer iteration to keep the line search stable and reduce data transfer. The native support of texture on GPU also allows fast sub-pixel interpolation, which provides higher accuracy.

The whole procedure of photometric tracking is summarized in Algorithm 3. The error term is the accumulative photometric and smoothness penalty error. The smoothness penalty is shrunk by a factor between $[0, 1]$. We find that 3 iterations and a shrunk factor of 0.9 lead satisfactory results for most scenarios.

5.3 Depth Estimation

Considering facial deformation can be semantically represented by facial expression variation, we reduce the dynamic depth reconstruction problem to a series of static ones for each individual expression. Since facial deformation

Algorithm 3: Photometric tracking

$I, E, R, t \leftarrow$ landmarks fitting;
while *error delta* $>$ *threshold* **do**
    Texture $\leftarrow I, E, R, t$;
    Smooth(E, R, t);
    Track(E, R, t);
    Shrink smoothness penalty;
    iteration $\leftarrow$ iteration $+ 1$;
    if *iteration* $>$ *max iteration* **then**
        break

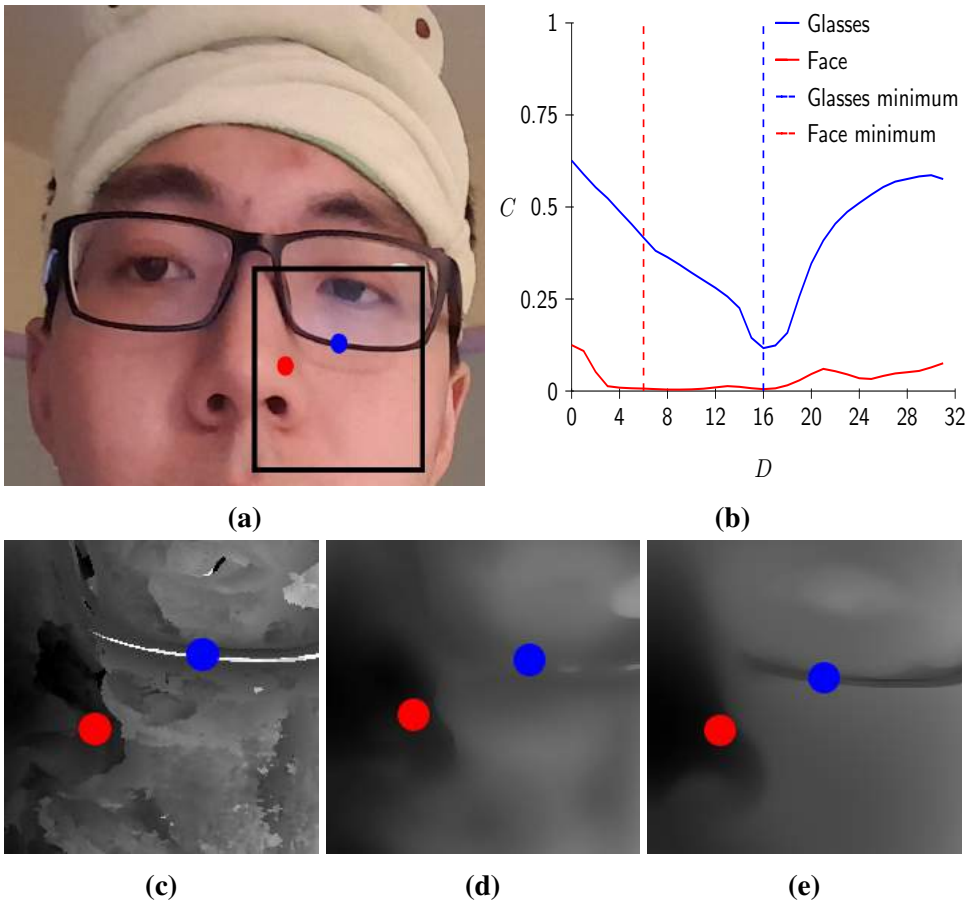


Figure 5.3: *Take the blue and red points on the reference image 5.3a for example, the red point on smooth varying face surface has multiple local cost volume minimums while the blue point on the glasses that protrude from the face has one clear minimum close to the true value as seen in 5.3b. The low quality of pixel intensity as the matching feature leads to ambiguous noisy local minimums in 5.3c. Moreover, as observed in the plot, a large range of depth values are not useful and therefore should not be searched during iteration. Evidently, if a general scheme is used such as the one in Newcombe et al. (2011b), the reconstructed depth in 5.3d can only resemble the essences of true object, whereas the depth in 5.3e reconstructed by the proposed scheme is more accurate.*

is expressed with blendshapes, the canonical rigid reference for each expression is established via clustering the blendshape coefficients.

For each cluster we select a source frame with most visible facial area, and the incoming frames are assigned with their corresponding clusters and used as photometric measurement target frame. The number of clusters is updated during tracking to reflect the fact that more expressions have been observed, which is performed by ensuring the standard deviation of a cluster is lower than a threshold. The choice of this threshold would influence the number of views required for each expression and the reconstruction quality, where we found that for the 25 dimension expression coefficient a standard deviation of 0.1 is a good compromise.

We denote the inverse of rotation R^{-1} and translation t^{-1} from source R_s and T_s to target image R_t and t_t as $R^{-1} = R_t^T \times R_s$ and $t^{-1} = t_t - R_t^T \times t_s$. The average photometric error $C(u, d)$ of pixel u in reference image F_r and target image F_t with the inverse depth d can be written as:

$$C_r(u, d) = \sum |F_r(u) - F_t(\Pi_A(A^{-1} \times [u, d], R^{-1}, t^{-1}))|. \tag{5.10}$$

Since the relative movement of face to the camera is small, the minimum average sum of the photometric error should correspond to the correct depth under the intensity consistent assumption. However, the per-pixel minimum might not necessarily be accurate or lead to smooth surface due to factors such as specular reflection, self occlusion and intensity ambiguity in texture-less areas.

In order to solve this, previous methods (Newcombe et al. 2011b; Graber et al. 2011, 2015) have employed a total variational minimization to remove noise. The goal is to minimize the gradient of depth map D to produce smooth surface and preserve depth discontinuity around edges, which is achieved via

minimizing

$$\min_{D} \int_{\Omega} |\nabla D(u)| + \lambda C(u, d), \tag{5.11}$$

where ∇ is the distributional derivative and Ω is the image domain. The variational term $|\nabla D|$ is convex whereas the data fidelity term $\lambda|C(u, d)|$ is non-convex. A convex approximation of the data fidelity term controlled by λ can be obtained by linearising the cost volume and solving the resulting approximation iteratively within a coarse-to-fine warping scheme. This would require keeping all the images thus significantly increasing the computational cost. Since the aim is to process long video sequences that contain as much expression and poses as possible, we follow the approach in Newcombe et al. (2011b), in which the energy functional is approximated by coupling the data and regularization terms through an auxiliary variable α (Chambolle and Pock 2011):

$$\min_{D,\alpha} \int_{\Omega} G|\nabla D(u)|_{\epsilon} + \lambda C(u, \alpha(u)) + \frac{1}{2\theta}\|D(u) - \alpha(u)\|, \tag{5.12}$$

Although L1 total variation is robust to outliers, it suffers from the staircase effect. One could alleviate this effect by applying Huber norm on the weighted variational term as $G|\nabla D|_{\epsilon}$, where $G = e^{-\nabla F}$ is the image gradient of the reference image computed from Equation 5.17, which is optionally normalized and scaled to reflect the smoothness regularization strength on edge boundaries. For continuous surface the Gaussian noise smaller than ϵ is smoothed by L2 norm while larger depth discontinuity are filtered by L1 norm.

Although the cost-volume is discrete, sub-sample refinement could be computed from performing a single Newton step using numerical differenti-

ation of the coupling term $E(u, d, \alpha) = \lambda C(u, \alpha) + \frac{1}{2\theta}\|D - \alpha\|$.

$$\bar{\alpha} = \alpha - \frac{\nabla E(u, d, \alpha)}{\nabla^2 E(u, d, \alpha)} \tag{5.13}$$

To produce a smooth surface, one limitation of such approximation is that the cost volume needs to be sampled at a very high rate with every possible depth. Note that a rough model of the face is readily available from the parametric model, it is used as a prior to accelerate the iteration and generate more accurate results. Based on this, several modifications are introduced to the original update scheme, which significantly speeds up the optimization. The effectiveness of our proposed scheme is shown in Fig. 5.3.

1. The search radius is set according to the photometric tracking error. Because detail not included in the parametric model is less likely to be correctly captured in the median texture, a larger search radius is used for pixels with bigger error. The search radius s is set to be positive correlated to sum of intensity difference between the synthesized rendering and the real images, and the search range is centred around the depth of face model r,

$$\alpha \in [r - s, r + s]. \tag{5.14}$$

2. When solving the auxiliary variable α in each iteration, if the absolute difference $|C(u, \alpha) - C(u, r)| < \epsilon$, the auxiliary variable α is set to r instead of performing the single Newton step refinement.

3. Assuming there is no major facial modification, the search radius is limited to the visible range $d < r$, where r is the depth value on the face model. For pixels not on the face model the search radius is set to the distance between the lowest and highest depth value of the face model.

   As the coupling energy term θ becomes larger, the feasible search range of auxiliary variable is shrunk as well. Given the cost volume minimum

and maximum of a pixel in current range, the coupling energy dictates that the solution should lie in the following bound,

$$C_u^{min} + \frac{1}{2\theta}\|D - \alpha\| \leq min(C_u^{max}, C(u,r)), \tag{5.15}$$

and the updated search radius is

$$s = 2\theta \cdot \lambda(min(C_u^{max}, C(u,r)) - C_u^{min}) \tag{5.16}$$

Following Stühmer et al. (2010), the duality principles leads us to the primal-dual form of Equation 5.12, where the primal variable is α and denote the dual variable is denoted as q. It is essential for the gradient operation ∇ that operates on the dual variable q to be different from the one that operates on image in Equation 5.9, in order for the Stokes theorem to hold exactly. The gradient of depth map D is computed with forward differences with Neumann boundary condition. The divergence of the dual variable q, which is the adjoint of the gradient of D, is computed with backward differences. For image of size (W, H), the numerical scheme is detailed as follows:

$$\begin{aligned}
\frac{\partial D(i,j)}{\partial x} &= \begin{cases} D(i+1,y) - D(i,j) & \text{if} 1 \leq i \leq W, \\ 0 & \text{otherwise.} \end{cases} \\
\frac{\partial D(i,j)}{\partial y} &= \begin{cases} D(i,j+1) - D(i,j) & \text{if} 1 \leq j \leq H, \\ 0 & \text{otherwise.} \end{cases}
\end{aligned} \tag{5.17}$$

$$
\begin{aligned}
div(p) = &\begin{cases} q_x(i,j) - q_x(i-1,j) & \text{if} 1 \leq i \leq W, \\ q_x(i,j) & \text{if} i = 1, \\ -q_x(i-1,j) & \text{otherwise.} \end{cases} \\
+ &\begin{cases} q_y(i,j) - q_y(i,j-1) & \text{if} 1 \leq j \leq H, \\ q_y(i,j) & \text{if} j = 1, \\ -q_y(i,j-1) & \text{otherwise.} \end{cases}
\end{aligned} \tag{5.18}
$$

Following the duality-based algorithm in Chambolle and Pock (2011), σ is selected as $\sigma = \tau\frac{1}{\mathcal{L}^2}$, $\mathcal{L} = 8$, $\theta = 1$ and $\tau = 0.01$. The Huber norm control variable ϵ is set based on the search grid of the cost volume. The dual and primal variable is minimized in an alternating manner by fixing one while solving for the other:

1. Fixed α, solve

$$
\min_{D} \int_{\Omega} |\nabla D|_{\epsilon} + \frac{1}{2\theta}\|D - \alpha\|. \tag{5.19}
$$

   The gradient ascent is performed on $\partial D = 0$, yielding

$$
\begin{aligned}
q^{n+1} &= \Psi(\frac{q + \sigma G \nabla D}{1 + \sigma\epsilon}), \\
D^{n+1} &= \frac{D^n + \tau(G \cdot div(q^{n+1}) + \frac{\alpha}{\theta})}{1 + \sigma\epsilon}.
\end{aligned} \tag{5.20}
$$

   where $\Psi(x) = \frac{x}{max(1,\|x\|)}$ is the resolvent operator that projects the gradient ascent step back onto the unit ball.

2. Fixed D, solve

$$
\min_{\alpha} \int_{\Omega} \frac{1}{2\theta}\|D - \alpha\| + \lambda C(u,\alpha), \tag{5.21}
$$

   which is achieved via point-wise exhaustive search with the aforementioned scheme.

Figure 5.4: *Our method is able to provide accurate 3D tracking which is crucial for successful depth estimation. Tracked 3D mesh is superimposed and the noisy 2D landmarks for initialization are shown in blue points.*

The point-wise search is independent of its neighbors and trivially parallelizable on modern GPU. The update for q and D on the other hand depends on its neighbors. Thus we use CUDA warp shuffles, which enable different processing units in the same wrap to share value through register and avoid reading/writing from global memory to reduce the overhead of syncing.

5.4 Experiments

In this section we detail the performance and implementation of our method. All of the experiments were done on a desktop PC with Intel Xeon (3.5 GHz), 32 GB RAM and GTX 980 graphics card. We designed two separate set of experiments to verify the effectiveness of our method. First we compare the facial performance tracking quality of our method to that of state-of-the-art facial landmark tracking methods in uncontrolled setting. Next we compare the depth estimation accuracy to SFM methods where the person remains still and the camera position changes.

The 2D landmark detection takes one millisecond to compute (Kazemi and Sullivan 2014). The surface parametrization is only performed once when the parametric model was initially being fitted to the 2D landmarks, which takes 100ms. For 1080p videos, OpenGL rendering takes 5ms, error evaluation and derivative computation taking 2ms and the smoothing operation takes less than 1ms. For depth map with a size of 600×800 pixels, the cost volume aggregation takes 5ms to execute with a search grid resolution of 64 levels. The tracking and photometric error computation runs on average around 35 fps. The denoising takes 100 iterations that finish within 110ms, and as a result the denoised depth map is generated per user request instantly.

Robust Tracking

To evaluate the tracking performance, we compare our method with existing facial landmark detection methods on the video dataset in Shen et al. (2015), as well as with our own recordings in tough situations, which are either downloaded from Youtube or recorded with a Samsung Galaxy S6 smart phone. Qualitative results are shown in Fig. 5.4, more of which can be found in the supplementary material.

The benchmark dataset 300 V-W (Shen et al. 2015) consists of videos recorded in uncontrolled environment with manually labeled landmark ground

Method	**Common Subset**	**Challenging Subset**	**Fullset**
ERT Kazemi and Sullivan (2014)	6.11	14.7	6.40
SDM Xiong and De (2013)	6.12	14.1	6.14
LBP Ren et al. (2014)	6.03	13.9	6.11
DDE Cao et al. (2014a)	5.45	11.9	6.32
Ours	**4.97**	**6.98**	**5.11**

Table 5.1: *The quantitative comparison with existing methods measured in averaged errors on the 300 V-W Shen et al. (2015), results taken from existing executable and literature. Note that both our method and Cao et al. (2014a) needs a few frames to start up, we excluded the results of first 3 seconds in each video.*

Method	**Average Error**	**Pose (s)**	**Depth (s)**
PMVS Furukawa and Ponce (2010)	1.4	25	283.4
GIPUMA Galliani et al. (2015)	1.1	25	95.9
Ours	**0.4**	**2.12**	**1.62**

Table 5.2: *The average error is computed from the squared error of the facial area to the Kinect Fusion scan. Results of PMVS (Furukawa and Ponce 2010) and GIPUMA (Galliani et al. 2015) are computed from 35 images, which are selected manually to cover most of the facial area and contain the least amount of motion blur. Results of our method are computed from 10s 30FPS short clips of the person. Example fused depth map of Furukawa and Ponce (2010); Galliani et al. (2015) are shown in Fig. 5.1*

.

truth. We redefine the landmark indices of the 3D parametric model according to the protocol in Shen et al. (2015). Comparative results with existing methods are illustrated in Table 5.1. Although our landmark detector is based on Kazemi and Sullivan (2014), which did not achieve the best result, building on top of its output our method achieved the best result on the challenging subset and fullset.

Depth Estimation

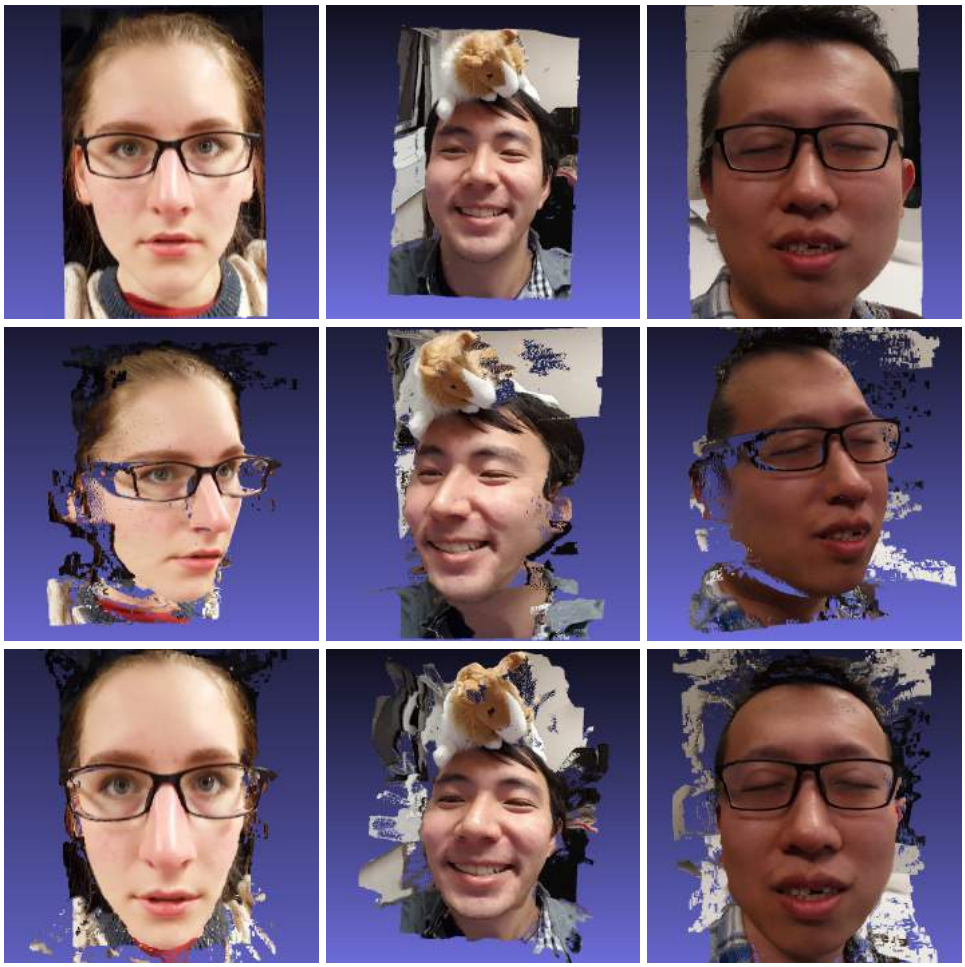


Figure 5.5: *From top to bottom: reference view, novel unseen views and perspective-aware portrait photos manipulation based on the depth map obtained from our method.*

Figure 5.6: *From top to bottom: reference view, novel unseen views and perspective-aware portrait photos manipulation based on the depth map obtained from our method.*

To evaluate the proposed depth estimation method we compared the reconstructed depth map of the same recording to that of Newcombe et al. (2011b), which took a similar approach for realtime general object reconstruction. The quantitative result is shown in Fig. 5.3d and 5.3e. At first glance, the depth map produced by Newcombe et al. (2011b) roughly captured essences of the face. However, closer inspection revealed that it failed to produce an equally accurate representation as the proposed method.

For quantitative comparison between our method and SFM methods (Furukawa and Ponce 2010; Galliani et al. 2015), we measure the average computation time and the RMSE (cm) error of the reconstructed depth map compared to the real physical face. It is shown in Table 5.2 that our method achieved the lowest error and need the least amount of time.

Inspired by Fried et al. (2016), we showcase novel views generated from the depth map reconstructed by our method as well as the perspective-aware portrait photos manipulation results in Fig. 5.6 and Fig. 5.5. Today most photos are taken using mobile devices with fixed focal length. With the high quality depth map, images captured by fixed focal length camera can be modified to simulate results captured with different focal length. More comprehensive results and dynamic examples can be found in the supplementary material.

5.5 Summary

We have proposed a novel method for dynamic 3D facial reconstruction from monocular videos in uncontrolled environments. The key contribution is undoing facial deformation via 3D facial performance tracking.

Experimental results show that our method is able to perform robust 3D facial tracking even from noisy output produced by the 2D landmark detector. Moreover, our method is able to produce realistic facial surface while preserving large facial geometry variation. Although our method only generates depth maps at the moment, we will investigate creating morphable 3D volumetric models for dynamic facial expression transfer and video retargeting in the future.

Chapter 6

Conclusions and Future Works

6.1 Conclusions

This thesis proposed 3D facial performance tracking from monocular RGB images in-the-wild. Chapter 2 gives an overview of existing works in related research areas. It discuss the current progress and limitation of recent works and investigate key challenges that need to be solved in order to obtain accurate results. In particular, 2D landmark detection, 3D facial performance tracking and 3D facial geometry reconstruction are discussed.

The first parts of 3D facial performance tracking are the detection of face bounding box and 2D facial landmark, which was discussed in Chapter 3. In uncontrolled environments, traditional 2D facial landmark detection methods often performs poorly, which affect the following 3D facial performance tracking negatively. Therefore in this chapter we proposed two robust facial landmark detection methods.

In the Section 3.1, we investigated the supervised descent method for 2D landmark detection and proposed a novel sign-correlation partition for global supervised descent method. It improves the performance of supervised descent method by computing better descent directions via sign-correlation partition. In Section 3.2, we proposed a novel 3D bilinear model for supervised descent method, as 2D based methods are inherently weaker than 3D

based methods on 3D object landmark detection such as facial landmark detection.

In the Chapter 4, we reconstruct the 3D facial geometry via solving a nonrigid alignment problem which fits 3D blendshape to the 2D landmarks produced from Chapter 3. The facial performance is tracked with person-specific 3D facial models. In this chapter we introduced several improvement over existing methods and demonstrated an efficient facial expression transfer application. In Section 4.1 we discuss landmark based 3D facial performance tracking and propose an efficient method for 3D facial expression transfer. In Section 4.2 we model the appearance of the actor and employ dense optical flow to automatically corrects the tracked results from landmark based tracking methods.

In Chapter 5 we introduce a dynamic 3D facial geometry reconstruction method for monocular RGB input. In order to obtain accurate 3D facial performance tracking results, the geometry of the actor will have to be modelled accurately as well. There exists many photogrammetry methods that are capable of reconstructing static objects with high accuracy. However in real world facial imagery capture scenarios people might be moving or have different expressions. To remove the interference from these factors we track the facial performance with a generic blendshapes from Chapter 4, and reconstruct the 3D geometry in realtime to generate more accurate person-specific 3D models.

In Section 5.2 of Chapter 5, we proposed a realtime dense 3D facial tracking method based on direct optimisation of the pose and expression parameter over the image domain via approximated gradient computed from the image gradient. In Section 5.3 we apply the realtime dense tracking method on videos and propose a novel depth reconstruction method to recover the 3D facial geometry in realtime.

6.2 Future Works

The quality of 3D facial performance tracking from monocular RGB input in-the-wild depends on multidisciplinary technology working together. Unlike facial performance tracking in studio settings, in real world uncontrolled environments there are many factors that could interfere with the tracking results, such as large facial appearance, pose, lighting variation and partial occlusion. Currently, many of the key parts of 3D facial performance tracking, such as facial landmark detection, 3D facial model creation and markerless tracking are still unsolved and under active research.

In terms of facial landmark detection, the recent advancement in deep learning has sparked many interests, which has lead to the publication of many exciting works. However, one common pet peeve of deep learning methods is that they require a large amount of computational resources, which makes them unsuitable for mobile devices. Since these deep learning methods have been proven to be incredibly successful for many applications, mobile device manufacturer has started developing hardware accelerated solutions. Integrating deep learning based methods into the 3D facial performance tracking pipeline would be promising since they have solved many computer vision problems.

From the perspective of 3D performance tracking, it would also be interesting to explore the end-to-end deep learning approach that directly predicts the pose, expression and identity from the image instead of explicitly solving for them given landmarks, or perform photometric optimisation. Additionally, it would be very useful to build better and more robust model to describe the appearance and physical traits of the human face to enable photometric optimisation in a wider range of environments.

Finally, the research on photogrammetry which reconstruct the 3D geometry of static objects from 2D RGB images will continue to be active because the resolution, speed and availability of RGB cameras are still far better than any depth sensors. For deformable objects reconstruction, although there

has been some recent works such as dynamic fusion, which aim to reconstruct nonrigidly deforming objects from RGB-D input, directly reconstructing these objects from RGB cameras still has not attracted much attention. This is due to the fact that there are too many unknowns for generic dynamic object reconstruction. However, since we have developed strong prior for human faces it would be interesting to find out how much it could help us tackle this challenge.

References

Abdel-Aziz, Y., 1971. Direct linear transformation from comparator coordinates in close-range photogrammetry. *In: ASP Symposium on Close-Range Photogrammetry in Illinois, 1971*.

Agarwal, S., Snavely, N., Seitz, S. M. and Szeliski, R., 2010. Bundle adjustment in the large. *In: European conference on computer vision*. Springer, 29–42.

Bardsley, D. and Li, B., . *3D Reconstruction Using the Direct Linear Transform with a Gabor Wavelet Based Correspondence Measure*. Technical report, Technical Report [online]. 2004,[cit. 2011-05-03]. Available from:¡ http://bardsley. org. uk/wp-content/uploads/2007/02/3dreconstruction-using-the-direct-linear-transform. pdf.

Barron, J. T. and Malik, J., 2015. Shape, illumination, and reflectance from shading. *IEEE transactions on pattern analysis and machine intelligence*, 37 (8), 1670–1687.

Basri, R., Jacobs, D. and Kemelmacher, I., 2007. Photometric stereo with general, unknown lighting. *International Journal of Computer Vision*, 72 (3), 239–257.

Beeler, T., Bickel, B., Beardsley, P., Sumner, B. and Gross, M., 2010. High-quality single-shot capture of facial geometry. *In: ACM Transactions on Graphics (TOG)*. ACM, volume 29, 40.

Beeler, T., Hahn, F., Bradley, D., Bickel, B., Beardsley, P., Gotsman, C., Sumner, R. W. and Gross, M., 2011. High-quality passive facial performance

capture using anchor frames. *In: ACM Transactions on Graphics (TOG)*. ACM, volume 30, 75.

Belhumeur, P. N., Jacobs, D. W., Kriegman, D. J. and Kumar, N., 2013. Localizing parts of faces using a consensus of exemplars. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 35 (12), 2930–40.

Blanz, V., Basso, C., Poggio, T. and Vetter, T., 2003. Reanimating faces in images and video. *In: Computer graphics forum*. Wiley Online Library, volume 22, 641–650.

Blanz, V. and Vetter, T., 1999. A morphable model for the synthesis of 3d faces. *In: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 187–194.

Bleyer, M., Rhemann, C. and Rother, C., 2011. Patchmatch stereo-stereo matching with slanted support windows. *In: Bmvc*. volume 11, 1–11.

Bonnans, J.-F., Gilbert, J. C., Lemaréchal, C. and Sagastizábal, C. A., 2006. *Numerical optimization: theoretical and practical aspects*. Springer Science & Business Media.

Booth, J., Roussos, A., Zafeiriou, S., Ponniah, A. and Dunaway, D., 2016. A 3d morphable model learnt from 10,000 faces. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5543–5552.

Breiman, L., 2001. Random forests. *Machine learning*, 45 (1), 5–32.

Brox, T., Bruhn, A., Papenberg, N. and Weickert, J., 2004. High accuracy optical flow estimation based on a theory for warping. *In: European conference on computer vision*. Springer, 25–36.

Brox, T. and Malik, J., 2011. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE transactions on pattern analysis and machine intelligence*, 33 (3), 500–513.

Bulat, A. and Tzimiropoulos, G., 2017. How far are we from solving the 2d &

3d face alignment problem?(and a dataset of 230,000 3d facial landmarks). *arXiv preprint arXiv:1703.07332*.

Burgos-Artizzu, X. P., Perona, P. and Dollár, P., 2013. Robust face landmark estimation under occlusion. *In: Proceedings of the IEEE International Conference on Computer Vision*. 1513–1520.

Burgosartizzu, X. P., Perona, P. and Dollar, P., 2013. Robust face landmark estimation under occlusion. *In: IEEE International Conference on Computer Vision*. 1513–1520.

Cao, C., Bradley, D., Zhou, K. and Beeler, T., 2015. Real-time high-fidelity facial performance capture. *ACM Transactions on Graphics (TOG)*, 34 (4), 46.

Cao, C., Hou, Q. and Zhou, K., 2014a. Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Transactions on Graphics (TOG)*, 33 (4), 43.

Cao, C., Weng, Y., Lin, S. and Zhou, K., 2013. 3d shape regression for real-time facial animation. *ACM Transactions on Graphics (TOG)*, 32 (4), 41.

Cao, C., Weng, Y., Zhou, S., Tong, Y. and Zhou, K., 2014b. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 20 (3), 413–425.

Cao, C., Wu, H., Weng, Y., Shao, T. and Zhou, K., 2016. Real-time facial animation with image-based dynamic avatars. *ACM Transactions on Graphics (TOG)*, 35 (4), 126.

Cao, X., Wei, Y., Wen, F. and Sun, J., 2014c. Face alignment by explicit shape regression. *International Journal of Computer Vision*, 107 (2), 177–190.

Chambolle, A. and Pock, T., 2011. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision*, 40 (1), 120–145.

Chen, L., Armstrong, C. W. and Raftopoulos, D. D., 1994. An investigation

on the accuracy of three-dimensional space reconstruction using the direct linear transformation technique. *Journal of biomechanics*, 27 (4), 493–500.

Chow, C. and Liu, C., 1968. Approximating discrete probability distributions with dependence trees. *IEEE transactions on Information Theory*, 14 (3), 462–467.

Coleman, T. F. and Li, Y., 1996. An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on optimization*, 6 (2), 418–445.

Cootes, T. F., Edwards, G. J. and Taylor, C. J., 1998. Active appearance models. *In: European conference on computer vision*. Springer, 484–498.

Cootes, T. F., Edwards, G. J., Taylor, C. J. et al., 1999. Comparing active shape models with active appearance models. *In: BMVC*. Citeseer, volume 99, 173–182.

Cootes, T. F., Edwards, G. J., Taylor, C. J. et al., 2001. Active appearance models. *IEEE Transactions on pattern analysis and machine intelligence*, 23 (6), 681–685.

Cootes, T. F., Ionita, M. C., Lindner, C. and Sauer, P., 2012. Robust and accurate shape model fitting using random forest regression voting. *In: European Conference on Computer Vision*. Springer, 278–291.

Cootes, T. F. and Taylor, C. J., 1992. Active shape modelssmart snakes. *In: BMVC92*, Springer, 266–275.

Cootes, T. F., Taylor, C. J., Cooper, D. H. and Graham, J., 1995. Active shape models—their training and application. *Computer Vision & Image Understanding*, 61 (1), 38–59.

Cootes, T. F., Wheeler, G. V., Walker, K. N. and Taylor, C. J., 2002. View-based active appearance models. *Image and vision computing*, 20 (9), 657–664.

Criminisi, A., Shotton, J. and Konukoglu, E., 2012. Decision forests: A unified framework for classification, regression, density estimation, manifold

learning and semi-supervised learning. *Foundations and Trends® in Computer Graphics and Vision*, 7 (2–3), 81–227.

Cristinacce, D. and Cootes, T. F., 2006. Feature detection and tracking with constrained local models. *BMVC*, 41, 929–938.

Cristinacce, D. and Cootes, T. F., 2007. Boosted regression active shape models. *In: BMVC*. volume 1, 7.

Curtis, M. L., 2012. *Matrix groups*. Springer Science & Business Media.

Dalal, N. and Triggs, B., 2005. Histograms of oriented gradients for human detection. *In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. IEEE, volume 1, 886–893.

Danelljan, M., Häger, G., Khan, F. and Felsberg, M., 2014. Accurate scale estimation for robust visual tracking. *In: British Machine Vision Conference, Nottingham, September 1-5, 2014*. BMVA Press.

Dantone, M., Gall, J., Fanelli, G. and Van Gool, L., 2012. Real-time facial feature detection using conditional regression forests. *In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2578–2585.

Davis, T. A., 2006. *Direct methods for sparse linear systems*. SIAM.

De Lathauwer, L., De Moor, B. and Vandewalle, J., 2000. A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications*, 21 (4), 1253–1278.

Dollar, P., Welinder, P. and Perona, P., 2010. Cascaded pose regression. *IEEE*, 238 (6), 1078–1085.

Donner, R., Reiter, M., Langs, G., Peloschek, P. and Bischof, H., 2006. Fast active appearance model search using canonical correlation analysis. *IEEE transactions on pattern analysis and machine intelligence*, 28 (10), 1690–1694.

Dovgard, R. and Basri, R., 2004. Statistical symmetric shape from shading

for 3d structure recovery of faces. *In: European Conference on Computer Vision*. Springer, 99–113.

Felzenszwalb, P. F. and Huttenlocher, D. P., 2005. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61 (1), 55–79.

Felzenszwalb, P. F. and Huttenlocher, D. P., 2006. Efficient belief propagation for early vision. *International journal of computer vision*, 70 (1), 41–54.

Feng, Z. H., Hu, G., Kittler, J., Christmas, W. and Wu, X. J., 2015a. Cascaded collaborative regression for robust facial landmark detection trained using a mixture of synthetic and real images with dynamic weighting. *IEEE Transactions on Image Processing*, 24 (11), 3425–3440.

Feng, Z. H., Huber, P., Kittler, J., Christmas, W. and Wu, X. J., 2015b. Random cascaded-regression copse for robust facial landmark detection. *IEEE Signal Processing Letters*, 22 (1), 76–80.

Fried, O., Shechtman, E., Goldman, D. B. and Finkelstein, A., 2016. Perspective-aware manipulation of portrait photos. *ACM Transactions on Graphics (TOG)*, 35 (4), 128.

Friedman, J., Hastie, T., Tibshirani, R. et al., 2000. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28 (2), 337–407.

Furukawa, Y. and Ponce, J., 2009. Accurate camera calibration from multi-view stereo and bundle adjustment. *International Journal of Computer Vision*, 84 (3), 257–268.

Furukawa, Y. and Ponce, J., 2010. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence*, 32 (8), 1362–1376.

Galliani, S., Lasinger, K. and Schindler, K., 2015. Massively parallel multiview stereopsis by surface normal diffusion. *In: Proceedings of the IEEE International Conference on Computer Vision*. 873–881.

Gao, X.-S., Hou, X.-R., Tang, J. and Cheng, H.-F., 2003. Complete solution

classification for the perspective-three-point problem. *IEEE transactions on pattern analysis and machine intelligence*, 25 (8), 930–943.

Garcia, D., 2010. Robust smoothing of gridded data in one and higher dimensions with missing values. *Computational statistics & data analysis*, 54 (4), 1167–1178.

Garrido, P., Valgaerts, L., Wu, C. and Theobalt, C., 2013. Reconstructing detailed dynamic face geometry from monocular video. *ACM Trans. Graph.*, 32 (6), 158–1.

Garrido, P., Zollhöfer, M., Casas, D., Valgaerts, L., Varanasi, K., Pérez, P. and Theobalt, C., 2016. Reconstruction of personalized 3d face rigs from monocular video. *ACM Transactions on Graphics (TOG)*, 35 (3), 28.

Ghiasi, G. and Fowlkes, C., 2014. Occlusion coherence: Localizing occluded faces with a hierarchical deformable part model. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2385–2392.

Gonzalez-Mora, J., De la Torre, F., Murthi, R., Guil, N. and Zapata, E. L., 2007. Bilinear active appearance models. *In: Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 1–8.

Gonzalezmora, J., Torre, F. D. L., Murthi, R., Guil, N. and Zapata, E. L., 2007. Bilinear active appearance models. *In: IEEE International Conference on Computer Vision*. 1–8.

Gotardo, P. F., Simon, T., Sheikh, Y. and Matthews, I., 2015. Photogeometric scene flow for high-detail dynamic 3d reconstruction. *In: Proceedings of the IEEE International Conference on Computer Vision*. 846–854.

Graber, G., Balzer, J., Soatto, S. and Pock, T., 2015. Efficient minimal-surface regularization of perspective depth maps in variational stereo. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 511–520.

Graber, G., Pock, T. and Bischof, H., 2011. Online 3d reconstruction using

convex optimization. *In: Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. IEEE, 708–711.

Gross, R., Matthews, I. and Baker, S., 2005. Generic vs. person specific active appearance models. *Image and Vision Computing*, 23 (12), 1080–1093.

Guenter, B., Grimm, C., Wood, D., Malvar, H. and Pighin, F., 1998. Making faces. *In: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. ACM, 55–66.

Hartigan, J. A. and Wong, M. A., 1979. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28 (1), 100–108.

Hartley, R. and Zisserman, A., 2003. *Multiple view geometry in computer vision*. Cambridge university press.

Hernandez, M., Choi, J. and Medioni, G., 2012. Laser scan quality 3-d face modeling using a low-cost depth camera. *In: Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*. IEEE, 1995–1999.

Hinton, G. E., Osindero, S. and Teh, Y.-W., 2006. A fast learning algorithm for deep belief nets. *Neural computation*, 18 (7), 1527–1554.

Hirschmuller, H., 2008. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30 (2), 328–341.

Horn, B. K. and Schunck, B. G., 1981. Determining optical flow. *Artificial intelligence*, 17 (1-3), 185–203.

Hsieh, P.-L., Ma, C., Yu, J. and Li, H., 2015. Unconstrained realtime facial performance capture. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1675–1683.

Huang, G. B., Mattar, M., Berg, T. and Learned-Miller, E., 2008. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments.

Huang, G. B., Ramesh, M., Berg, T. and Learned-Miller, E., 2007. Labeled

faces in the wild: A database for studying face recognition in unconstrained environments.

Huang, J., Shi, X., Liu, X., Zhou, K., Wei, L.-Y., Teng, S.-H., Bao, H., Guo, B. and Shum, H.-Y., 2006. Subspace gradient domain mesh deformation. *In: ACM Transactions on Graphics (TOG)*. ACM, volume 25, 1126–1134.

Ichim, A. E., Bouaziz, S. and Pauly, M., 2015. Dynamic 3d avatar creation from hand-held video input. *ACM Transactions on Graphics (ToG)*, 34 (4), 45.

Igarashi, T., Moscovich, T. and Hughes, J. F., 2005. As-rigid-as-possible shape manipulation. *In: ACM transactions on Graphics (TOG)*. ACM, volume 24, 1134–1141.

Jackson, A. S., Bulat, A., Argyriou, V. and Tzimiropoulos, G., 2017. Large pose 3d face reconstruction from a single image via direct volumetric cnn regression. *arXiv preprint arXiv:1703.07834*.

Jolliffe, I., 2002. Principal component analysis.

Kazemi, V. and Sullivan, J., 2014. One millisecond face alignment with an ensemble of regression trees. *In: Computer Vision and Pattern Recognition*. 1867–1874.

Kemelmacher-Shlizerman, I. and Basri, R., 2011. 3d face reconstruction from a single image using a single reference face shape. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33 (2), 394–405.

Kholgade, N., Matthews, I. and Sheikh, Y., 2011. Content retargeting using parameter-parallel facial layers. *In: Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 195–204.

Kim, K. I., Jung, K. and Kim, H. J., 2002. Face recognition using kernel principal component analysis. *Signal Processing Letters, IEEE*, 9 (2), 40–42.

Kolda, T. G. and Sun, J., 2008. Scalable tensor decompositions for multi-

aspect data mining. *In: Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. IEEE, 363–372.

Kostinger, M., Wohlhart, P., Roth, P. M. and Bischof, H., 2011. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. *In: IEEE International Conference on Computer Vision Workshops, ICCV 2011 Workshops, Barcelona, Spain, November 6-13, 2011*. 2144–2151.

Le, V., Brandt, J., Lin, Z., Bourdev, L. and Huang, T. S., 2012a. Interactive facial feature localization. *In: European Conference on Computer Vision*. Springer, 679–692.

Le, V., Brandt, J., Lin, Z., Bourdev, L. and Huang, T. S., 2012b. Interactive facial feature localization. *In: European Conference on Computer Vision*. 679–692.

Lee, H. S. and Kim, D., 2009. Tensor-based aam with continuous variation estimation: application to variation-robust face recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 31 (6), 1102–16.

Li, D.-H. and Fukushima, M., 2001. A modified bfgs method and its global convergence in nonconvex minimization. *Journal of Computational and Applied Mathematics*, 129 (1), 15–35.

Li, H., Trutoiu, L., Olszewski, K., Wei, L., Trutna, T., Hsieh, P.-L., Nicholls, A. and Ma, C., 2015. Facial performance sensing head-mounted display. *ACM Transactions on Graphics (TOG)*, 34 (4), 47.

Li, H., Yu, J., Ye, Y. and Bregler, C., 2013. Realtime facial animation with on-the-fly correctives. *ACM Trans. Graph.*, 32 (4), 42–1.

Lienhart, R. and Maydt, J., 2002. An extended set of haar-like features for rapid object detection. *In: Image Processing. 2002. Proceedings. 2002 International Conference on*. IEEE, volume 1, I–900.

Liu, D. C. and Nocedal, J., 1989. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45 (1-3), 503–528.

Liu, S., Yang, X., Wang, Z., Xiao, Z. and Zhang, J., 2016. Real-time facial expression transfer with single video camera. *Computer Animation and Virtual Worlds*, 27 (3-4), 301–310.

Liu, S., Zhang, Y., Yang, X., Shi, D. and Zhang, J. J., 2017. Robust facial landmark detection and tracking across poses and expressions for in-the-wild monocular video. *Computational Visual Media*, 3 (1), 33–47.

Lowe, D. G., 1999. Object recognition from local scale-invariant features. *In: Computer vision, 1999. The proceedings of the seventh IEEE international conference on*. Ieee, volume 2, 1150–1157.

Lowe, D. G., 2004. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60 (2), 91–110.

Luo, P., Wang, X. and Tang, X., 2012. Hierarchical face parsing via deep learning. *In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2480–2487.

Ma, W.-C., Jones, A., Chiang, J.-Y., Hawkins, T., Frederiksen, S., Peers, P., Vukovic, M., Ouhyoung, M. and Debevec, P., 2008. Facial performance synthesis using deformation-driven polynomial displacement maps. *In: ACM Transactions on Graphics (TOG)*. ACM, volume 27, 121.

Markuš, N., Frljak, M., Pandžić, I. S., Ahlberg, J. and Forchheimer, R., 2013. Object detection with pixel intensity comparisons organized in decision trees. *arXiv preprint arXiv:1305.4537*.

Masi, I., Rawls, S., Medioni, G. and Natarajan, P., 2016. Pose-aware face recognition in the wild. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4838–4846.

Moré, J. J., 1978. The levenberg-marquardt algorithm: implementation and theory. *In: Numerical analysis*, Springer, 105–116.

Murphy, K. P., Weiss, Y. and Jordan, M. I., 1999. Loopy belief propagation for approximate inference: An empirical study. *In: Proceedings of*

the Fifteenth conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc., 467–475.

Newcombe, R. A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A. J., Kohi, P., Shotton, J., Hodges, S. and Fitzgibbon, A., 2011a. Kinectfusion: Real-time dense surface mapping and tracking. *In: Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*. IEEE, 127–136.

Newcombe, R. A., Lovegrove, S. J. and Davison, A. J., 2011b. Dtam: Dense tracking and mapping in real-time. *In: Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2320–2327.

Ojala, T., Pietikäinen, M. and Harwood, D., 1996. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29 (1), 51–59.

Pandzic, I. S. and Forchheimer, R., 2003. *MPEG-4 facial animation: the standard, implementation and applications*. John Wiley & Sons.

Papandreou, G. and Maragos, P., 2008. Adaptive and constrained algorithms for inverse compositional active appearance model fitting. *In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 1–8.

Patel, A. and Smith, W. A., 2009. Shape-from-shading driven 3d morphable models for illumination insensitive face recognition. *In: BMVC*. 1–10.

Paysan, P., Knothe, R., Amberg, B., Romdhani, S. and Vetter, T., 2009. A 3d face model for pose and illumination invariant face recognition. *In: Advanced Video and Signal Based Surveillance, 2009. AVSS'09. Sixth IEEE International Conference on*. Ieee, 296–301.

Rall, L. B., 1981. Automatic differentiation: Techniques and applications.

Ramanan, D., 2015. Face detection, pose estimation, and landmark localization in the wild. *In: IEEE Conference on Computer Vision and Pattern Recognition*. 31–37.

Ranjan, R., Sankaranarayanan, S., Castillo, C. D. and Chellappa, R., 2017. An all-in-one convolutional neural network for face analysis. *In: Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on*. IEEE, 17–24.

Ren, S., Cao, X., Wei, Y. and Sun, J., 2014. Face alignment at 3000 fps via regressing local binary features. *IEEE Transactions on Image Processing*, 1685–1692.

Sagar, M., 2006. Facial performance capture and expressive translation for king kong. *In: ACM SIGGRAPH 2006 Sketches*. ACM, 26.

Sagonas, C., Tzimiropoulos, G., Zafeiriou, S. and Pantic, M., 2013a. 300 faces in-the-wild challenge: The first facial landmark localization challenge. *In: Proceedings of the IEEE International Conference on Computer Vision Workshops*. 397–403.

Sagonas, C., Tzimiropoulos, G., Zafeiriou, S. and Pantic, M., 2013b. A semi-automatic methodology for facial landmark annotation. *In: IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 896–903.

Saito, S., Li, T. and Li, H., 2016. Real-time facial segmentation and performance capture from rgb input. *In: European Conference on Computer Vision*. Springer, 244–261.

Saragih, J. and Goecke, R., 2007. A nonlinear discriminative approach to aam fitting. *In: 2007 IEEE 11th International Conference on Computer Vision*. IEEE, 1–8.

Saragih, J. M., Lucey, S. and Cohn, J. F., 2009. Probabilistic constrained adaptive local displacement experts. *In: Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*. IEEE, 288–295.

Scherbaum, K., Ritschel, T., Hullin, M., Thormählen, T., Blanz, V. and Seidel, H.-P., 2011. Computer-suggested facial makeup. *In: Computer Graphics Forum*. Wiley Online Library, volume 30, 485–492.

Shen, J., Zafeiriou, S., Chrysos, G. G., Kossaifi, J., Tzimiropoulos, G. and Pantic, M., 2015. The first facial landmark tracking in-the-wild challenge: Benchmark and results. *In: Proceedings of the IEEE International Conference on Computer Vision Workshops*. 50–58.

Shi, F., Wu, H.-T., Tong, X. and Chai, J., 2014. Automatic acquisition of high-fidelity facial performances using monocular videos. *ACM Transactions on Graphics (TOG)*, 33 (6), 222.

Shi, J., Samal, A. and Marx, D., 2006. How effective are landmarks and their geometry for face recognition? *Computer Vision and Image Understanding*, 102 (2), 117–133.

Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Smith, B. M. and Zhang, L., 2012. Joint face alignment with non-parametric shape models. *In: European Conference on Computer Vision*. Springer, 43–56.

Smolyanskiy, N., Huitema, C., Liang, L. and Anderson, S. E., 2014. Real-time 3d face tracking based on active appearance model constrained by depth data. *Image and Vision Computing*, 32 (11), 860–869.

Snavely, N. et al., 2010. Bundler: Structure from motion (sfm) for unordered image collections. *Available online: phototour. cs. washington. edu/bundler/(accessed on 12 July 2013)*, 1.

Sorkine, O., Cohen-Or, D., Lipman, Y., Alexa, M., Rössl, C. and Seidel, H.-P., 2004. Laplacian surface editing. *In: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*. ACM, 175–184.

Stühmer, J., Gumhold, S. and Cremers, D., 2010. Real-time dense geometry from a handheld camera. *In: Joint Pattern Recognition Symposium*. Springer, 11–20.

Sturm, P., 2014. Pinhole camera model. *In: Computer Vision*, Springer, 610–613.

Sun, Y., Wang, X. and Tang, X., 2013. Deep convolutional network cascade for facial point detection. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3476–3483.

Suwajanakorn, S., Kemelmacher-Shlizerman, I. and Seitz, S. M., 2014. Total moving face reconstruction. *In: European Conference on Computer Vision*. Springer, 796–812.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A., 2015. Going deeper with convolutions. *In: Proceedings of the IEEE conference on computer vision and pattern recognition*. 1–9.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z., 2016. Rethinking the inception architecture for computer vision. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2818–2826.

Tarrés, F. and Rama, A., 2012. Gtav face database. *GVAP, UPC*.

Taylor, C. J. and Kriegman, D. J., 1994. Minimization on the lie group so (3) and related manifolds. *Yale University*, 16, 155.

Thies, J., Zollhöfer, M., Nießner, M., Valgaerts, L., Stamminger, M. and Theobalt, C., 2015. Real-time expression transfer for facial reenactment. *ACM Transactions on Graphics (TOG)*, 34 (6), 183.

Thies, J., Zollhofer, M., Stamminger, M., Theobalt, C. and Nießner, M., 2016. Face2face: Real-time face capture and reenactment of rgb videos. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2387–2395.

Tong, Y., Liu, X., Wheeler, F. W. and Tu, P. H., 2012. Semi-supervised facial landmark annotation. *Computer Vision and Image Understanding*, 116 (8), 922–935.

Tresadern, P. A., Sauer, P. and Cootes, T. F., 2010. Additive update predictors in active appearance models. *In: BMVC*. Citeseer, volume 2, 4.

Tzimiropoulos, G. and Pantic, M., 2013. Optimization problems for fast aam fitting in-the-wild. *In: Proceedings of the IEEE international conference on computer vision*. 593–600.

Tzimiropoulos, G. and Pantic, M., 2014. Gauss-newton deformable part models for face alignment in-the-wild. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1851–1858.

Valgaerts, L., Wu, C., Bruhn, A., Seidel, H.-P. and Theobalt, C., 2012. Lightweight binocular facial performance capture under uncontrolled lighting. *ACM Trans. Graph.*, 31 (6), 187.

Van Ginneken, B., Frangi, A. F., Staal, J. J., Romeny, B. M. and Viergever, M. A., 2002. Active shape model segmentation with optimal features. *medical Imaging, IEEE Transactions on*, 21 (8), 924–933.

Viola, P. and Jones, M. J., 2004. Robust real-time face detection. *International journal of computer vision*, 57 (2), 137–154.

Vlasic, D., Brand, M., Pfister, H. and Popović, J., 2005. Face transfer with multilinear models. *In: ACM Transactions on Graphics (TOG)*. ACM, volume 24, 426–433.

Wang, N., Gao, X., Tao, D. and Li, X., 2014. Facial feature point detection: A comprehensive survey. *arXiv preprint arXiv:1410.1037*.

Wang, Y., Lucey, S. and Cohn, J. F., 2008. Enforcing convexity for improved alignment with constrained local models. *In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 1–8.

Weise, T., Bouaziz, S., Li, H. and Pauly, M., 2011. Realtime performance-based facial animation. *In: ACM Transactions on Graphics (TOG)*. ACM, volume 30, 77.

Weise, T., Li, H., Van Gool, L. and Pauly, M., 2009. Face/off: Live facial pup-

petry. *In: Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer animation*. ACM, 7–16.

Weng, Y., Cao, C., Hou, Q. and Zhou, K., 2014. Real-time facial animation on mobile devices. *Graphical Models*, 76 (3), 172–179.

Williams, L., 1990. Performance-driven facial animation. *In: ACM SIGGRAPH Computer Graphics*. ACM, volume 24, 235–242.

Wu, C., Bradley, D., Gross, M. and Beeler, T., 2016. An anatomically-constrained local deformation model for monocular face capture. *ACM Transactions on Graphics (TOG)*, 35 (4), 115.

Wu, H., Liu, X. and Doretto, G., 2008. Face alignment via boosted ranking model. *In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 1–8.

Wu, Y., Wang, Z. and Ji, Q., 2013. Facial feature tracking under varying facial expressions and face poses based on restricted boltzmann machines. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3452–3459.

Xing, J., Niu, Z., Huang, J., Hu, W. and Yan, S., 2014. Towards multi-view and partially-occluded face alignment. *In: Computer Vision and Pattern Recognition*. 1829–1836.

Xiong, X. and De, F., la Torre, 2013. Supervised descent method and its applications to face alignment. *In: IEEE Conference on Computer Vision & Pattern Recognition*. 532–539.

Xiong, X. and De la Torre, F., 2015. Global supervised descent method. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2664–2673.

Yan, J., Lei, Z., Yi, D. and Li, S. Z., 2013a. Learn to combine multiple hypotheses for accurate face alignment. *In: IEEE International Conference on Computer Vision Workshops*. 392–396.

Yan, J., Zhang, X., Lei, Z., Yi, D. and Li, S. Z., 2013b. Structural models for

face detection. *In: Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*. IEEE, 1–6.

Yang, F., Wang, J., Shechtman, E., Bourdev, L. and Metaxas, D., 2011. Expression flow for 3d-aware face component transfer. *In: ACM Transactions on Graphics (TOG)*. ACM, volume 30, 60.

Yang, H., He, X., Jia, X. and Patras, I., 2015a. Robust face alignment under occlusion via regional predictive power estimation. *IEEE Transactions on Image Processing*, 24 (8), 2393–403.

Yang, H., Jia, X., Patras, I. and Chan, K. P., 2015b. Random subspace supervised descent method for regression problems in computer vision. *IEEE Signal Processing Letters*, 22 (10), 1816–1820.

Yin, L., Wei, X., Sun, Y., Wang, J. and Rosato, M. J., 2006. A 3d facial expression database for facial behavior research. *In: 7th international conference on automatic face and gesture recognition (FGR06)*. IEEE, 211–216.

Zafeiriou, S., Trigeorgis, G., Chrysos, G., Deng, J. and Shen, J., 2017. The menpo facial landmark localisation challenge: A step towards the solution. *In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*. IEEE, 2116–2125.

Zhang, L., Chu, R., Xiang, S., Liao, S. and Li, S. Z., 2007. Face detection based on multi-block lbp representation. *In: Advances in biometrics*, Springer, 11–18.

Zhang, L., Snavely, N., Curless, B. and Seitz, S. M., 2008. Spacetime faces: High-resolution capture for˜ modeling and animation. *In: Data-Driven 3D Facial Animation*, Springer, 248–276.

Zhang, Y., Liu, S., Yang, X., Shi, D. and Zhang, J. J., 2016. Sign-correlation partition based on global supervised descent method for face alignment. *In: Asian Conference on Computer Vision*. Springer, 281–295.

Zhang, Y., Liu, S., Yang, X., Zhang, J. and Shi, D., 2017. Supervised co-

ordinate descent method with a 3d bilinear model for face alignment and tracking. *Computer Animation and Virtual Worlds*, 28 (3-4).

Zhang, Z., Luo, P., Loy, C. C. and Tang, X., 2014a. Facial landmark detection by deep multi-task learning. *In: European Conference on Computer Vision*. Springer, 94–108.

Zhang, Z., Luo, P., Loy, C. C. and Tang, X., 2014b. Facial landmark detection by deep multi-task learning. *In: European Conference on Computer Vision*. 94–108.

Zhao, C., Cham, W.-K. and Wang, X., 2011. Joint face alignment with a generic deformable face model. *In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 561–568.

Zhao, X., Chai, X. and Shan, S., 2012. Joint face alignment: Rescue bad alignments with good ones by regularized re-fitting. *In: Computer Vision–ECCV 2012*, Springer, 616–630.

Zhou, F., Brandt, J. and Lin, Z., 2013. Exemplar-based graph matching for robust facial landmark localization. *In: Proceedings of the IEEE International Conference on Computer Vision*. 1025–1032.

Zhou, S. K. and Comaniciu, D., 2007. Shape regression machine. *In: Biennial International Conference on Information Processing in Medical Imaging*. Springer, 13–25.

Zhu, S., Li, C., Loy, C. C. and Tang, X., 2015. Face alignment by coarse-to-fine shape searching. *In: CVPR*. 4998–5006.

Zhu, X., Lei, Z., Liu, X., Shi, H. and Li, S. Z., 2016. Face alignment across large poses: A 3d solution. *In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 146–155.

Zhu, X. and Ramanan, D., 2012. Face detection, pose estimation, and landmark localization in the wild. *In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2879–2886.