DOI: 10.1111/cgf.14463
EUROGRAPHICS 2022 / R. Chaine and M. H. Kim
(Guest Editors)

Volume 41 (2022), Number 2

Compact Facial Landmark Layouts for Performance Capture

E. Zell[1,2] and R. McDonnell[1]

[1]Trinity College Dublin
[2]University of Bonn

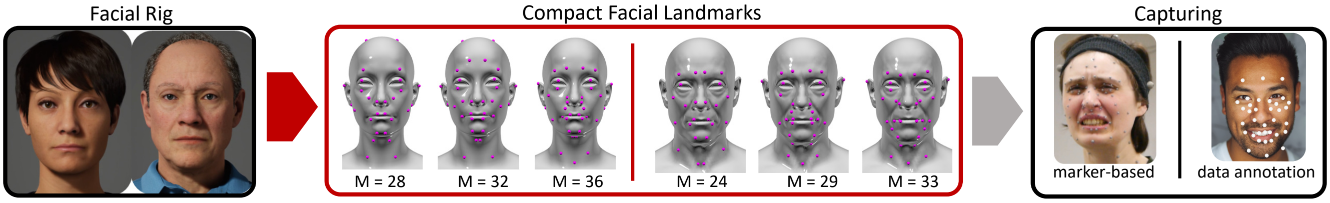


Figure 1: *Different to previous work, we suggest to derive facial landmarks from a low-dimensional facial rig by analyzing the degrees of freedom. Our method (red) is purely based on the existing animation model and does not require large character databases or person-specific 4D sequences. Different compact layouts are computed by out method for two of Epic's Metahuman character, with* $\varepsilon = 0.3$, 0.5 *and* 0.7 *for the female and* $\varepsilon = 0.5$, 0.7 *and* 0.8 *for the male character.*

Abstract

An abundance of older, as well as recent work exists at the intersection of computer vision and computer graphics on accurate estimation of dynamic facial landmarks with applications in facial animation, emotion recognition, and beyond. However, only a few publications exist that optimize the actual layout of facial landmarks to ensure an optimal trade-off between compact layouts and detailed capturing. At the same time, we observe that applications like social games prefer simplicity and performance over detail to reduce the computational budget especially on mobile devices. Other common attributes of such applications are predefined low-dimensional models to animate and a large, diverse user-base. In contrast to existing methods that focus on creating person-specific facial landmarks, we suggest to derive application-specific facial landmarks. We formulate our optimization method on the widely adopted blendshape model. First, a score is defined suitable to compute a characteristic landmark for each blendshape. In a following step, we optimize a global function, which mimics merging of similar landmarks to one. The optimization is solved in less than a second using integer linear programming and guarantees a globally optimal solution to an NP-hard problem. Our application-specific approach is faster and fundamentally different to previous, actor-specific methods. Resulting layouts are more similar to empirical layouts. Compared to empirical landmarks, our layouts require only a fraction of landmarks to achieve the same numerical error when reconstructing the animation from landmarks. The method is compared against previous work and tested on various blendshape models, representing a wide spectrum of applications.

1. Introduction

Over the last two dacades, facial animation capturing evolved from a research topic relevant only for high-end VFX application to a widely accessible technology and is nowadays even integrated in smartphones. Current applications span from highly-detailed captures of digital doubles to simple emoji animation, and from highly actor-specific solutions to a nearly unlimited user base. Besides capturing technology, best practices evolved for character creation pipelines paving the way for parametric character configurators like Epic's MetaHuman, Daz3D Genesis or Polywink. The originally linear workflow, starting with motion capturing and move afterwards to character creation and animation retargeting became more and more non-linear due to convenient access and compelling prices of pre-built characters. But if the character to animate exists before the actual capturing, is it possible to limit the capturing data and minimize data and processing time? We investigate the question of how to distinguish between relevant and non-relevant information for capturing facial animation *before* the actual capturing session without requiring a large database of facial expressions.

Previous work demonstrated that facial deformation can be reconstructed from a sparse set of feature markers [BBA*07,SLS*12, LZD13]. This observation inspired a significant number of research works on accurate detection of facial landmarks [DYOY18] as well as its use in applications like face reconstruction [LZZL16], facial tracking [CHZ14], emotion recognition [BZLM18] or as a preprocessing step in facial re-enactment [TZS*16]. In the times of deep-learning accurate facial landmark detection remains an integral part of various recent publications, e.g. [CVW19, ZDKZ19, FHCP19, YLC*19, GPKZ19], demonstrating that robust estimation of sparse feature points is of high relevance and interest. In contrast to the high number of publications on accurate facial landmark detection, we identified only two academic publications [LZD13, CXZ*02] that optimize the layout of facial landmarks. Interestingly, to the best of our knowledge, optimization methods have been neglected for practical layouts of facial landmarks. Instead, previous work and existing annotated image-data define the layouts of facial landmarks in subsequent publications, especially for data-driven methods. But, what if the facial landmarks we track have been badly chosen despite our best intentions? Or is it possible to speed-up data annotation time, without loosing accuracy?

Defining a layout of facial landmarks or markers for sparse facial tracking is a trade-off between capturing as much detail as possible and focusing only on relevant information. In fact, the approach is comparable to lossy data compression methods that allow a strong compression ratio, but when pushed too far, the decompression may break. In this work, we aim to identify the limits of compact landmark layouts for facial performance capturing. We observe that in many practical applications the character to animate is different in appearance to the original actor and a mapping is required of the actor's performance into a low-dimensional space of blendshape weights or rig controllers. This mapping makes compression and information loss inevitable. At the same time, defining the boundary of relevant and non-relevant information becomes easy. Information that is lost after the low-dimensional mapping could be neglected before capturing. In this paper, our key insight is that compact facial landmarks layouts can be defined by considering the degrees of freedom of the final animation model (Fig. 1). It is therefore possible to predict custom facial landmarks (Fig. 2) only with data that is already part of the animation pipeline and without any additional overhead like a dense capture of the actor [LZD13] or a statistical model [LBB*17].

Blendshape interpolation is the dominant approach for facial animation on many consumer devices (e.g., iPhoneX) and in professional context [Sey16], where the number of blendshapes defines the degrees of freedom of the model. We assume that the computation of the blendshape weights from a sparse set of facial landmarks is sufficiently constrained if at least one representative landmark exists for each blendshape. Suitable criteria for considering a landmark as representative are: good visibility (ideally during all possible expressions), and strong relative displacement compared to all other vertex displacements within the same blendshape. Placing landmarks at vertices with strong displacements ensures also placement at salient positions of the dynamic expression. In practice, a single landmark can even represent the motion of several blendshapes at the same time, e.g., closing and opening the eyelid. Thus, by placing the landmarks carefully, the minimal set of landmarks will be smaller than the degrees of freedom of the blendshape model. In our work, we consider every vertex of the 3D blendshape model as a possible candidate for a landmark. To solve the landmark placement problem numerically, we first estimate whether a vertex can represent different blendshapes or not (Section 3.1). In a second step, a minimal set of landmarks is computed that represents different blendshapes (Section 3.2). The general problem is equivalent to the NP-hard weighted set cover problem. Despite being NP-hard, computing a global minimum takes less than a second using a state-of-the-art integer programming solver, which is significantly faster than [CXZ*02, LZD13] previous methods. Our method is evaluated on several professional blendshape models consisting of up to 237 blendshapes, and compared against existing facial landmark layouts. In addition, we discuss several practical extensions like symmetrical layouts, optional backup landmarks for cases where more than one representative landmark should exist, and we compute layouts for the FLAME/3D FACS dataset [LBB*17, CKH11]. A reference implementation of our method will is published with the paper.

2. Related Work

Most existing facial landmark layouts have been derived empirically by placing landmarks either uniformly [KMS11, MJC*08], at locations with strong displacements [Wil90, CET98], at salient positions of the face (MPEG-4) or a combination of these three criteria [BBA*07]. The layout of most active appearance models (AAM) consist of 68 facial landmarks [MB04, GMC*10] and captures eyelids, eyebrows, lips, the nose and the chin. In earlier work [CET98, HLZC01] the sampling density of facial landmarks ranged between 72-122, but the layout was nearly identical. This layout remained largely unchanged until now because of annotated image datasets, e.g. [AZCP13, KS14, WGJ17, DYOY18]. To overcome the limitations of incompatible layouts, Sagonas et al. [SAT*16] developed an semi-supervised annotation tool and transferred different layouts [Mar98, KWRB11, BJKK11, MMK*99, LBL*12] to a 68 facial landmarks layout. Other notable variations to the 68 facial landmarks layout are, e.g. [TKC01, CHZ14], where additional landmarks are added at the cheeks and nasolabial folds (wrinkles between nose and mouth corners). The landmark layouts computed by our method are within the variation range of empirically derived layouts. At the same time, our layouts show that the chin and nose area are often over-sampled, and that cheeks or nasolabial folds should be included. We observe also that empirical landmark layouts tend to follow salient features of the static faces (e.g., tip of the nose), rather than dynamic facial features.

Facial landmarks or markers also play an important role within the computer graphics domain, since the pioneering work of facial capturing with retro-reflective facial markers [Wil90]. Despite the advantages of markerless facial capturing [ZTG*18], professional commercial solutions, e.g. VICON Cara or CubicMotion as well as in-house tools in big VFX companies [SML16] still often operate with visually distinctive features like dark points to compensate for tangential drift in featureless areas like cheeks.

While clear recommendations are given by [BB14] on which locations are best to capture the rigid motion of a head, a clear answer is missing for dynamic faces. An often encountered strategy to compute sparse landmark layouts is to first reduce the dimensions of a dense motion dataset using PCA [CXZ*02, LZWM06, WJZ13], or a blendshape model [RGL15]. In a second step, the rows of the PCA matrix are either clustered (k-means) or re-weighted to identify a representative set of landmarks. Please note that every vertex of the 3D model is associated with a row of the PCA matrix. Unfortunately, k-means clustering of high-dimensional vectors (3× number of principle components) is known to be error-prone [NJW01] and is confirmed in our evaluation. Le et al. [LZD13], optimize marker layouts for thin shell deformation, to approximate dense 4D sequences of facial performances. Unfortunately, in our experiments the suggested block coordinate descent algorithm turned out to be error prone to local minima and depends strongly on the initial landmark placement. Furthermore, facial animation based on shape-interpolation is beyond the scope of their method (see Section 4.2. in [LZD13]).

Optimizing marker layouts has also been addressed for capturing the body and hands by improving the condition number of the underlying inverse kinematics matrix [SWR*18]. The equivalent matrix for blendshapes is the delta-blendshape matrix [LAR*14] which contains all blendshape displacement without the neutral expression. In our tests we found that vertex displacements of one blendshape tend to have similar directions and differ only by the magnitude. In consequence, re-assigning a landmark to another vertex will change the magnitude, but not the direction of the column vectors and thus the condition number barely changes.

In this work, we focus on blendshapes, which is probably the most frequent shape-interpolation method for facial expressions in practice. Other local shape decomposition methods have been suggested in the past [TDlTM11, BBB*14] and variations of our method are especially suitable for shape decomposition methods where the local deformation differ spatially [NVW*13, TGL*18]. Sparse feature points can also serve as a representation basis in the latent space of neural networks [WCL*19, WML21], however this direction is beyond the scope our our work, but offers a promising direction for future work.

Concurrent to our work, quadratic integer optimization [KS21] was suggested to optimize UI layouts for facial rigs. While the objectives of the two works is different, our method computes more compact layouts (Fig. 1) on Epic's Metahuman rigs (ca. 30 landmarks vs 142 parameters). In addition, our linear formulation does not require pre-processing operations like clustering to reduce the number of vertex candidates and is magnitudes faster (1s vs 162-553s).

To conclude, existing methods compute sparse landmark layouts by sparsifing a dense 4D capture and thus compute a person-specific layout. Apart from being personalized, the computed layouts lack similarity with generalized, empirically derived layouts. In contrast to previous work, landmarks computed by our application-specific method highly correlate with semantically meaningful locations, achieving closer resemblance to empirical layouts. Our optimization computes a global optimum, and is often magnitudes faster.

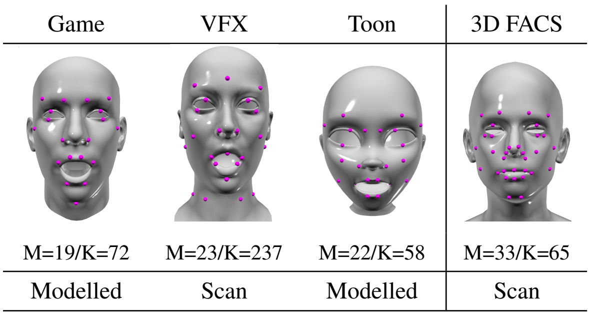


Figure 2: *Left: Compact layouts of facial landmarks for specific animation models. Results are given for* $\varepsilon = 0.5$*. Right: A minimalistic layout derived from a FACS [EF78] model based on our evaluation results (Section 4.2). M and K represent the number of landmarks and the number of blendshapes of the animation model respectively.*

3. Minimalistic Facial Layouts

A blendshape model consists of a polygon mesh with a neutral expression and K blendshapes. The set of vertex positions for the neutral expressions is $\{\mathbf{v}_1^0, \ldots, \mathbf{v}_N^0\}$, with n being the vertex index. The vertex displacement between blendshape k and the neutral expression is defined as: $d\mathbf{v}_n^k = \mathbf{v}_n^k - \mathbf{v}_n^0$. All vertex displacements of a single blendshape form a delta-blendshape $\{d\mathbf{v}_1^k, \ldots, d\mathbf{v}_N^k\}$. In our work, we optimize for a set of M feature points or 3D landmarks $\mathcal{F} = \{\mathbf{f}_1, \ldots, \mathbf{f}_M\}$. On the one hand we aim to find a set with the smallest number of landmarks. On the other hand, it must be guaranteed that facial motion will be reconstructed reliably from the 3D landmarks.

Our proposed optimization method consists of two steps. First, we identify potential candidates for facial landmarks and compute their *quality* to reconstruct the captured facial motion (Section 3.1). In a second step, we identify the best combination of all candidates (Section 3.2), both in terms of the smallest number of facial landmarks as well as their overall potential to reconstruct the facial animation. Finally, we discuss practical extensions (Section 3.3) to accomplish nicer layouts and improve robustness for performance estimation.

3.1. Suitable Landmark Candidates

Starting with the general assumption that every vertex of the blendshape model could be a potential facial landmark, we aim to quantify the *quality* of suitable candidates. Traditionally, the deformation of a blendshape only affects parts of the face. In consequence, the weight for blendshape k can be only computed from landmarks where the vertex displacement is $d\mathbf{v}_n^k > 0$. We save the information whether a landmark is a suitable candidate for a blendshape or not in a set of boolean variables $\{c_n^1, \ldots, c_n^K\}$ with $c_n^k \in \{0, 1\}$.

In addition, we save the *quality* of all vertices in a variable ω_n^k. By *quality* we mean any numerical properties that predict how easy the facial landmark can be captured or how robust is the computation of blendshape weights from 3D landmarks. After testing various metrics to estimate the *quality* ω_n^k of a facial landmark, which is

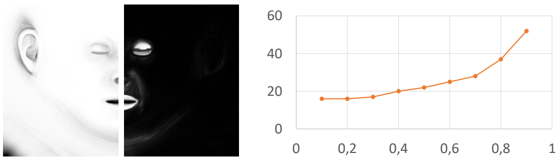


Figure 3: *Left: Ambient occlusion texture (cropped) and the difference between the pre-computed global visibility and the ambient occlusion of the neutral expression. Right: The plotted number of landmarks vs ε relationship.*

discussed in Section 2 and Section 3.1.3, we identified global visibility l_n (Section 3.1.2) and relative vertex displacement r_n^k (Section 3.1.1) as the most consistent and generic (Fig. 3). Both metrics are normalized, to ensure consistent results across different characters and vertex numbers. Based on our experiments, equal weighting ($\alpha = 0.5$) is recommended. Examples for layouts computed with different quality metrics are shown in Fig. 4.

$$\omega_n^k = c_n^k(2-(1-\alpha)r_n^k - \alpha l_n), \quad 0 \le r_n^k, l_n \le 1 \tag{1}$$

3.1.1. Relative Displacement

In order to maximize the reconstruction of the animation, landmarks should be placed where the vertex displacement is largest for the individual blendshape. Independent of the tracking accuracy of the landmark, the reconstruction error will be as small as possible if we maximize the signal-to-noise-ratio. To account for blendshapes with strong as well as subtle displacements, we preferred a relative vertex displacement measure r_n^k for a specific blendshape k as a *quality* metric for estimating the signal-to-noise-ratio.

$$r_n^k = \frac{\left\| d\mathbf{v}_n^k \right\|}{\max\left\{ \left\| d\mathbf{v}_1^k \right\|, \dots, \left\| d\mathbf{v}_N^k \right\| \right\}}, \quad 0 \le r_n^k \le 1 \tag{2}$$

$\max\left\{ \left\| d\mathbf{v}_1^k \right\|, \dots, \left\| d\mathbf{v}_N^k \right\| \right\}$ denotes the maximum vertex displacement of blendshape k along all vertices N. When computing the total quality of a landmark in (Eq. 1), we negate our relative displacement weights, to comply with our optimization function (Eq. 5).

In addition, we define the variable c_n^k based on the relative vertex displacement r_n^k. c_n^k saves the information whether a vertex n fulfills the minimal criteria for being a suitable landmark candidate for a blendshape k.

$$c_n^k = \begin{cases} 0 & r_n^k < \varepsilon \\ 1 & r_n^k \ge \varepsilon \end{cases} \tag{3}$$

If r_n^k is bigger than a threshold ε, this vertex is considered as a possible candidate for reconstructing the weights of blendshape k. Theoretically, $0 < \varepsilon \le 1$, but based on our evaluation (Fig. 3, Section 4) we recommend $0.3 \le \varepsilon \le 0.7$, depending whether accurate reconstruction or a small set of landmarks is preferred.

3.1.2. Global Visibility

Occlusion is a well-known limitation when capturing the position of facial landmarks. Predicting in advance whether a landmark will be occluded during capturing is extremely challenging. However, given the blendshape model, we have a proxy for the range of motion of the human face. Furthermore, we can assume that landmarks are placed at semantically identical positions and expressions remain similar between the captured and animated face, e.g. a smile should remain a smile. We take advantage of this information to approximate the visibility of all possible candidates based on the blendshape model.

Because the relative position of the camera(s) with respect to the actor's face is unknown in advance we compute the global visibility of a vertex. We check for possible self-occlusions by placing a normal aligned hemisphere at the vertex and sample along all directions of the unit hemisphere. The concept is equivalent to computing ambient occlusion for every vertex in the context of rendering [PJH16]. The luminance L_n^k for blendshape k and vertex n computes the ratio of visible rays along the hemisphere. To estimate the visibility of vertices across the entire range of motion of the face, ambient occlusion is computed for every vertex and every blendshape separately. From a capturing perspective we want to avoid the worst-case scenario, where landmarks are completely lost due to occlusion, therefore the smallest L_n^k is selected in the end for each vertex (Fig. 3).

$$l_n = \min\left\{ L_n^0, \dots, L_n^K \right\} \tag{4}$$

In scenarios with known camera positions, like head-mounted cameras, the visibilty term greatly simplifies. Instead of sampling across the entire vertex-hemisphere, a single visibility check between the vertex and the camera is sufficient. The remaining computations remain the same.

3.1.3. Discarded quality metrics

We tested SIFT features or perceptual mesh saliency [CSPF12]. In our experiments these metrics created a high percentage of false positives. SIFT [Low04] features emphasize person specific features (e.g. moles) that scale poorly across different identities. Or the features are located at object silhouettes which are view-dependent and non-stationary with respect to the surface. Mesh saliency [CSPF12] which is largely based on mean curvature and Gaussian curvatures [Rus04] preferred undesired landmark locations inside of folds or inside the nose. We also discovered that other intuitive salient features, like the tip of the nose or the left and right corners, are misleading as their deformation is relatively small during facial motions. Other metrics like the condition number did not improved out results further (see Section 2). Please notice that previous work [CXZ*02,LZD13,RGL15] on facial landmark computation is only based on vertex displacements.

3.2. Minimalistic Set of Landmarks

After refining the potential candidates for suitable landmarks, we obtain two sets for each vertex n: (i) $\{c_n^1, \dots, c_n^K\}$ defines whether this vertex fulfills the minimal criteria to represent a blendshape k or not, and (ii) $\{\omega_n^1, \dots, \omega_n^K\}$ defines the *quality* of representing a specific blendshape. Out of all landmark candidates, we search for a combination with the smallest number of landmarks and good reconstruction properties of the facial animation. This problem is a

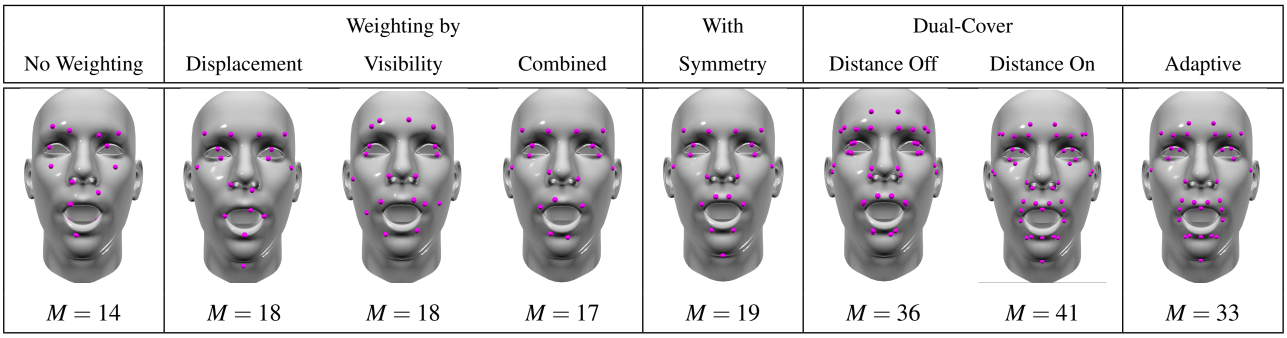


Figure 4: *Visual comparison of layouts for the Game model with* 72 *blendshapes and* $\varepsilon = 0.5$. *M represents the number of landmarks.*

variation of the NP-hard set cover problem [Kar72] which can be formulated as a boolean linear program.

$$\min_{f_n} \sum_{n=1}^{N} f_n \sum_{k=1}^{K} \omega_n^k\,, \qquad \text{with}\, \omega_n^k \geq 0 \qquad (5a)$$

$$\text{s.t.} \sum_{n=1}^{N} c_n^k f_n \geq 1\,, \qquad \text{for all}\, k \in K \qquad (5b)$$

$$f_n \in \{0,1\}\,, \qquad \text{for all}\ n \in N \qquad (5c)$$

The unknown boolean variables f_n specify whether a vertex will be selected as a landmark or not and our objective is to minimize the function 5a. In addition, K linear constraints (Eq. 5b) enforce that at least one suitable landmark is selected for each blendshape. If we ignore for a second the importance weighting factor ω_n^k, Eq. 5a would search for a landmark set with the smallest number of landmarks. By adding the importance factor ω_n^k, we favor landmark locations with greater visibility and larger relative displacements. Consequently, these landmarks are easier to track and allow more robust reconstruction of the facial animation. Compared to the unweighted case, the downside of individual importance weights is that the overall number of landmarks may increase slightly. Once all f_n are computed, the final landmark set is identified by the index n. All selected landmarks form the set $\mathcal{F} = \{\mathbf{f}_1, \ldots, \mathbf{f}_M\}$. Fig. 4 visualizes the influence of the different weighting terms on the final landmark placement and Fig. 5 the energy distribution of a single vertex.

In contrast to previous work [CXZ*02, LZWM06, LZD13, SWR*18] we do not set the number of landmarks directly. This eliminates the need, e.g. in k-means clustering, to test different numbers of clusters to identify the best separation. Instead the number of landmarks is part of our minimalistic landmark optimization that, by construction, guarantees that a suitable landmark exists to control each blendshape. The number of landmarks is indirectly controlled by setting ε (Fig. 3) and by modifying the importance weight ω_n^k (Eq. 1, 3).

3.2.1. Solver

Despite being NP-hard, a global optimal solution is computed in only a few seconds using state-of-the-art mixed integer solvers like CPLEX. Besides the number of vertices and blendshapes, the actual blendshapes have an impact on the solver's performance. Current integer solvers reduce the search space in advance by selecting vertices with the smallest importance weight if several vertices have identical sets of landmark candidates $\{c_n^1, \ldots, c_n^K\}$.

3.3. Practical Extensions

Existing facial landmark layouts tend to reflect the left and right symmetry of the face. In the following, we discuss how to achieve symmetric facial landmark layouts even for non-symmetric faces and the placement of additional landmarks beyond the minimal requirements.

3.3.1. Symmetry

To achieve symmetric layouts (Fig. 4, middle right), some pre-processing of the blendshapes is required. First, we compute symmetric vertex pairs between the left and right side of the face. Second, we compare the vertex displacements between the left and right side of the face. The bigger vertex displacement $d\mathbf{v}_n^k$ is assigned to the vertex of the right side of the face, while the left side is set to 0. This procedure is performed for every blendshape k separately. After pre-processing the blendshapes, the minimalistic landmark layout (Eq. 5) is computed as described previously and at the end, all landmarks are mirrored to the left side.

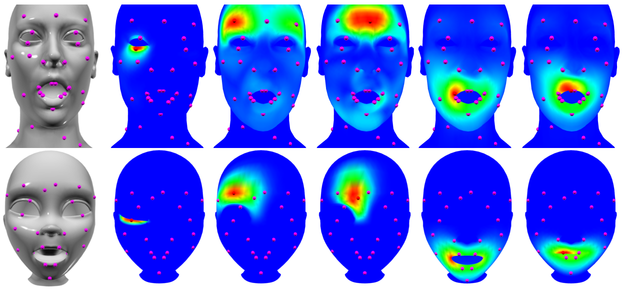


Figure 5: *Importance weights (normalized) for a selected landmark on the VFX (top) and Toon (bottom) models. Similar color encoding on nearby vertices indicate that small shifting of landmark location may still create nearly optimal layouts.*

3.3.2. Adaptive Multi-Cover Sets

The minimalistic layout, computed in the previous section, only guarantees one representative landmark for each of the blendshapes. Due to unpredictable occlusions (e.g. by a person's hand) or to improve reconstruction accuracy, it might be sensible to include more landmarks for each blendshape. Extending a set-cover problem to a multi-cover problem by setting $\sum_{n=1}^{N} c_n^k f_n \geq 2$ in Eq. 5b is trivial. Unfortunately, the result is unsatisfactory, as two nearby vertices will be selected instead of one (Fig. 4, "Distance Off"). In addition, we tested refining the initial layout using particle swarm optimization [SWR*18] in combination with a energy term maximizing the distance between two landmarks. Landmarks are placed better than for the trivial multi-cover formulation, but also a bit random. We therefore recommend to run the single-cover optimization (Section 3.2) multiple times and update at each iteration the set of potential landmark candidates as well as blendshapes requiring additional landmarks (see Algorithm 1, and Fig. 4, "Distance On")). Advantages of this approach are: minimal implementation overhead, fast run-times and possible adaptive oversampling, where the space constraints are sufficient.

After computing suitable candidates for the landmarks (Eq. 3) and their respective quality (Eq. 1), the following changes are required to the single-cover algorithm:

1. Remove vertices with assigned landmarks from the set of potential candidates by setting $\{c_n^1, \dots, c_n^K\} = 0$.
2. Remove vertices within the neighborhood of assigned landmarks from the set of potential candidates by setting $\{c_n^1, \dots, c_n^K\} = 0$
3. Exclude all blendshapes from Eq. 5 where no potential candidate for suitable landmarks exists or that have sufficient landmarks.
4. Compute the minimalistic layout by solving Eq. 5 and add the computed landmarks to $\mathcal{F}$.

All enumerated items are encapsulated in a for-loop (see Algorithm 1), which is executed until no suitable landmark candidate exists for any blendshape, or a maximum number of landmarks per blendshape is reached.

In our experiments we notice that one representative landmark leads to accurate reconstructions of expressions where blendshapes are distinctive (e.g. eyes, eyebrows) but not so, if many similar blendshape have large overlapping areas (e.g. mouth). To balance between minimalistic layouts and good reconstruction quality of the facial performance, we recommend that statistically one unique landmark should not be assigned to more than 10% of blendshapes and more landmarks should be added where this condition is violated (Fig. 4, "Adaptive"). Such a heuristic can be added in the adaptive multi-cover algorithm when the set of blendshapes is reduced (Algorithm 1, l.13).

Algorithm 1 Adaptive Multi-Cover

```
 1: $\mathcal{F} = \emptyset$                                      ▷ final landmarks
 2: $c_n^k =$ Eq. 3                                     ▷ suitable candidates (boolean)
 3: $\bar{\mathcal{K}} = \{1, \dots, K\}$                          ▷ blendshapes missing landmarks
 4: $i_{max}$                                           ▷ max number of landmarks/blendshape
 5: for $i = 1, \dots, i_{max} || \bar{\mathcal{K}} = \emptyset$ do
 6:     for $n = 1, \dots, N$ do                        ▷ update suitable candidates
 7:         if $\mathbf{v}_n \in \mathcal{F}$ then                         ▷ discard landmarks
 8:             $\{c_n^1, \dots, c_n^K\} = 0$
 9:         end if
10:         if $\mathbf{v}_n$ close to $\mathcal{F}$ then                ▷ discard nearby landmarks
11:             $\{c_n^1, \dots, c_n^K\} = 0$
12:         end if
13:     end for
14:     for $k = \{1, \dots, K\}$ do                    ▷ update set of blendshapes
15:         if $\{c_1^k, \dots, c_N^k\} = 0$ then
16:             remove $k$ from $\bar{\mathcal{K}}$
17:         end if
18:     end for
19:     $\mathcal{F}$ += Eq. 5 w.r.t. $c_n^k$ and $\bar{\mathcal{K}}$
20: end for
```

4. Evaluation

Data We test our algorithm on four representative blendshape models (Fig. 2) and two of Epic's Metahuman characters (Fig. 1). Our selection covers scan based and hand-modelled characters, various application cases (VFX, games, computer vision), stylizations (realistic and cartoon), as well as moderate to high numbers of blendshapes (up to 237). One advantage of our method is that it is suitable for non-realistic characters as long these follow human motion principles and proportions. Additional visual examples are provided in the supplemental material.

4.1. Comparison to Previous Work

We compare our landmark placement to all dedicated methods for facial landmark placement that we are aware of, which are surprisingly few. Cohen et al. [CXZ*02] cluster the rows of the PCA matrix and Reverdy et al. [RGL15] the delta-blendshape matrix. Le et al. [LZD13] minimize the error of a surface deformation model and in Sorkine and Cohen-Or [SC04] landmarks are placed where the reconstruction error of the surface deformation model is largest. All previous methods take vertex displacement into account. Some approaches support symmetric layouts. Landmark visibility is neglected by all previous methods. For a fair comparison, optimal landmark layouts are computed only based on vertex displacement for all test-cases. In addition, all vertices that have no displacement across all blendshapes ($\varepsilon < 0.01$) have been removed from the optimizations. In our implementations we followed closely the cited publications and recommended settings. Testing revealed that most methods converged well despite different initialization values. Only the results of Le et al. [LZD13] were prone to local minima and the final results were highly sensitive to the initial landmark placement, making it difficult to quantify the performance (Fig. 6).

The computation time of all algorithms largely depends on the number of linear systems to solve. For our method, we solve one boolean linear system (CPLEX, 0.63s), Cohen et al. [CXZ*02] compute a singular value decomposition (Eigen, 33s). Sorkine and Cohen-Or [SC04] solve M times a sparse linear system (Eigen, overall 22s), while for Le et al. [LZD13] the same sparse linear system is solved ten times M, where ten is the recommended num-

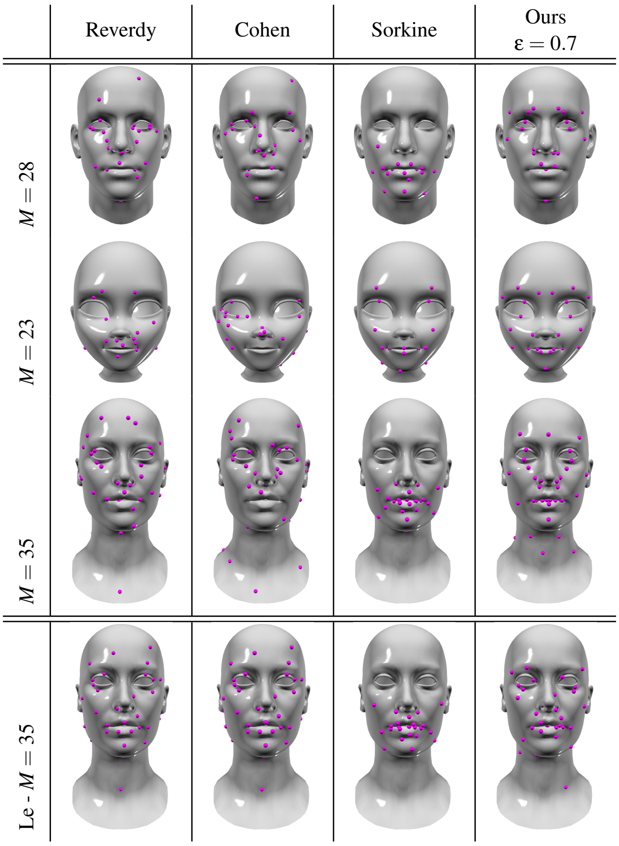


Figure 6: *Comparison of layouts computed by different methods based on vertex displacement only on (top) Game, (middle) Toon and (bottom) VFX models. Number of landmarks M is constant in each row.*

Last row shows the results of Le for different initialization setups (row above). Our method computes the most symmetric and balanced layouts even without additional constraints. Some landmarks may be hidden from camera view.

ber of iterations (Eigen, overall 165s). All listed timings were measured for the VFX-model and a computer with an Intel i7-8550U CPU. Visual examples of the computed layouts are provided for all methods in Fig. 6.

Statistical Data Statistical analysis can quantify well the worst case scenarios within a set of landmarks. Scenario 1: Landmarks are placed where no blendshape has a strong displacement. Such landmarks will have only little impact on the final animation, e.g., computing blendshape weights for the lip blendshapes from landmarks placed at the cheeks. Scenario 2: A blendshape does not have any landmarks within the area of moderate or strong displacement and therefore will not be activated during animation. To quantify existing and computed layouts, we set $\varepsilon = 0.3$ for a *moderate* cutoff value. For completeness, we also add a *strong* cutoff value of $\varepsilon = 0.7$, however, the implications are slightly different. A small number of landmarks or blendshapes for $\varepsilon = 0.7$ means that landmarks are extremely well placed. It does not indicate anymore that bad cases exist. Fig. 7 shows diagrams for all test-cases. Please note that methods based on clustering [CXZ*02, RGL15] place 10-20% of the landmarks where no blendshape has a moderate or strong displacement (Fig. 7).

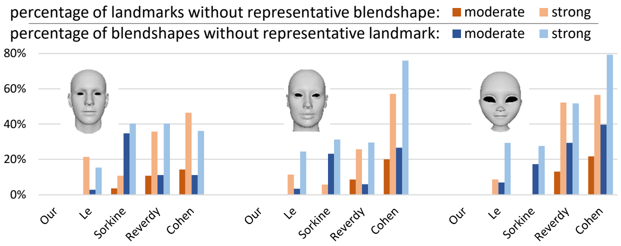


Figure 7: *Orange: The percentage of landmarks located at vertices where no blendshape has a strong* ($\varepsilon > 0.7$) *or moderate* ($\varepsilon > 0.3$) *displacement. Captured animation will be transferred inaccurately for these landmarks after retargeting. Blue: The number of blendshapes that have no landmarks located at vertices with strong* ($\varepsilon > 0.7$) *or moderate* ($\varepsilon > 0.3$) *displacements.*

In the case of existing facial landmark layouts, such as AR [Mar98], AFLW [KWRB11], LFPW [BJKK11], XM2VTS [MMK*99] or even the common 68 facial landmarks of AAMs (MultiPIE [GMC*10]), we observe that 15-30% of landmarks provide little or no information for facial animation (Fig. 8, left/orange bars). In addition, a representative landmark for (almost) each blendshape exist only for layouts with high numbers of landmarks (LFPW, XM2VTS, MultiPIE, MPEG-4). Common placement of unused landmarks in existing layouts are: ears and chin, corners of eyelids, and the ridge of the nose. We also observe substantial oversampling for the chin, nose and eyebrows, while cheeks and nasal folds are largely ignored. In contrast, our layouts achieve good statistical scores with substantially less landmarks.

Ground Truth Comparisons For ground truth comparisons, a blendshape model is animated first by activating individual blendshapes or pairs of blendshapes. In a second step, landmarks are extracted for different layouts and the original motion is reconstructed by solving a non-negative least-square problem [LAR*14]. The reconstruction error is smallest for our method among all existing optimization methods (Fig. 9). Comparing the results of all methods, we notice that especially if a representative landmark for one blendshape is missing, the expressions are not well reconstructed. Notice that this observation is equivalent to leaving one out tests. Compared to existing empirical models, we achieve visually indistinguishable results with only 38% of the landmarks (26 vs. 68). The root-mean-square error (RMSE) is on par with empirical layouts, while using only 56% of the landmarks (39 vs 68). The ablation study (Fig. 11) shows further that if we remove landmarks, but still try to keep the maximum number of representative landmarks, the reconstruction error will be smallest. The supplemental material provides additional comparisons, where we simulate inaccurate landmark placement by shifting the landmarks (supple-

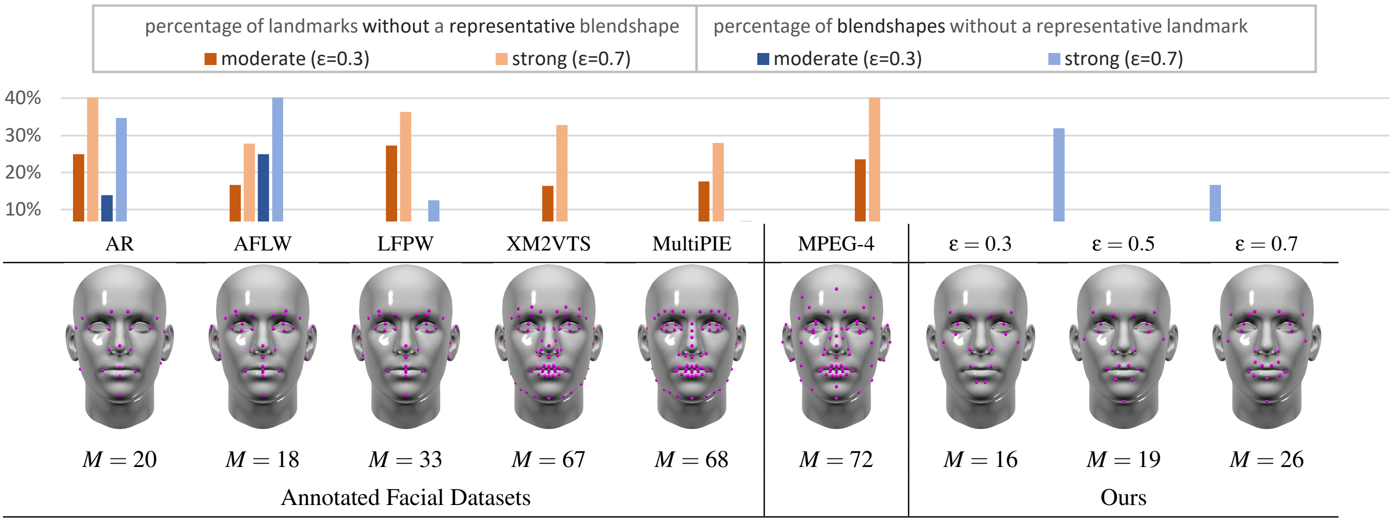


Figure 8: *Top: Statistical data for different layouts shown below. Bottom: Visual comparison of landmark layouts used in annotated computer vision datasets, the MPEG-4 standard, and our method (symmetric layout) for the Game model. In existing layouts, about 15-30% of landmarks are located were no blendshapes have at least moderate ($\varepsilon > 0.3$) displacement. At the same time, our layouts provide substantial information to all blendshapes, but use only 50% of the landmarks of existing layouts.*

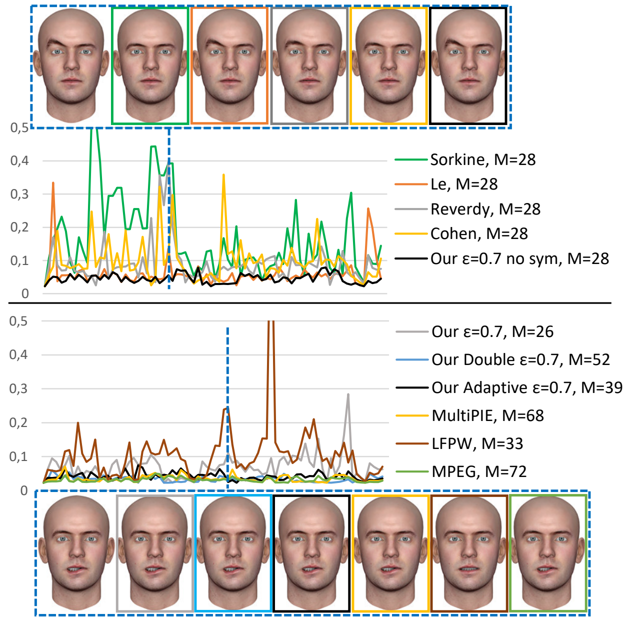


Figure 9: *Root-mean-square error (RMSE) in mm between ground truth and reconstructed animation. Reconstructed expressions are shown for the marked frame (dotted line).*

mental video). While this introduces small numerical errors, the reconstructed animation is still accurate.

4D Data Retargeting To compare different landmark layouts with real data we retarget to one 4D sequence of Zhang et al. [ZSCS04] consisting of different combinations of facial expressions. In a preprocessing step we register non-rigidly [LSP08] the neutral expression of the male blendshape model from Fig. 1 to the neutral expression of the 4D sequence. In a second step, we establish dense one-to-one correspondences between the neutral expressions and transfer the topology of the blendshape model to the entire 4D sequence. Furthermore, personalized blendshapes for the 4D model are created using deformation transfer [SP04]. The dataset consists of 384 frames and 157 blendshapes, all of equal topology and identical vertex set. For the evaluation, we computed landmark layouts with our single cover method and dedicated algorithms [CXZ*02,SC04,LZD13,RGL15] all with 27 landmarks (see Fig. 12). Because the method of Le et al. [LZD13] turned out to be prone to local minima and strongly depends on the initial landmark placement, three different initialization layouts [CXZ*02, SC04, RGL15] were tested. For further comparison, we computed two adaptive layouts where each blendshape has up to two representative landmarks. For the evaluation, the blendshape model was fitted towards the 4D sequence by minimizing the distance between the respective sparse landmarks. Our results show that, the root-mean-square error (RMSE), computed for all frames and vertices between the reconstructed expression and the 4D se-

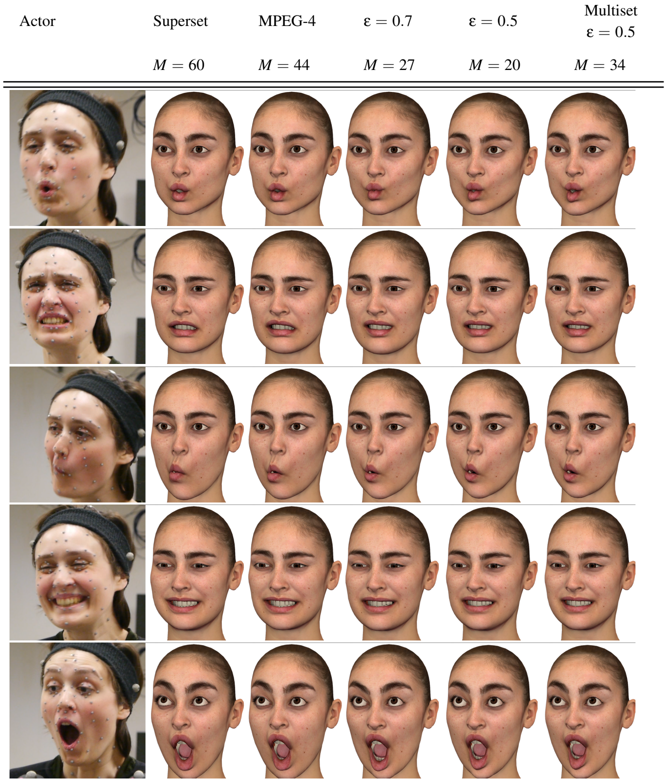


Figure 10: *Top: Visual comparison of the actor and reconstructed animation from facial landmarks on the VFX model. The superset is the combination of all layouts. For the MPEG-4 layout only 44 out of 72 marker could be captured because either the vertices were static (ears), markers were impossible to place (inner lips), or remained hidden due to strong self-occlusion markers (eyelids). Despite using only a fraction of landmarks, expressions are reconstructed faithfully. Eyes were animated procedurally. Examples are (roughly) ordered from top to bottom by increasing intensity of facial expressions.*

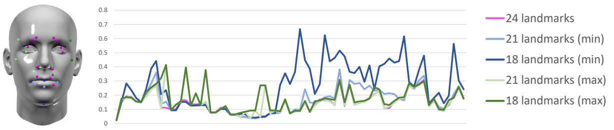


Figure 11: *Ablation study: We compare the reconstruction accuracy on our compact layout with* $\varepsilon = 0.7$ *and* $M = 24$*, against layouts where three or six landmarks have been removed. Landmark layouts are shown on the left. Layouts annotated with max, cover the maximum number of landmarks (*$M = 21$ *- without light blue landmarks,* $M = 18$ *- without dark and light blue landmarks), while layouts annotated with min, cover the smallest number of landmarks (*$M = 21$ *- without light green landmarks,* $M = 18$ *- without dark and light green landmarks).*

quence is smallest for our layout. Interestingly, the reconstruction error for Le et. al [LZD13] is worse than the initialization layout in all cases. Landmark layouts using our adaptive multi-cover algorithm had comparable to smaller RMSE. Comparison between the single cover ($\varepsilon = 0.7$, $M = 27$) and the adaptive multi-cover ($\varepsilon = 0.5$, $M = 46$) results indicates that a high ε is more beneficial than having additional landmarks in combination with a smaller ε. Overall, the results support our initial hypothesis from Section 3.1.1 on the importance of relative blendshape displacements for accurate recovery of facial expressions.

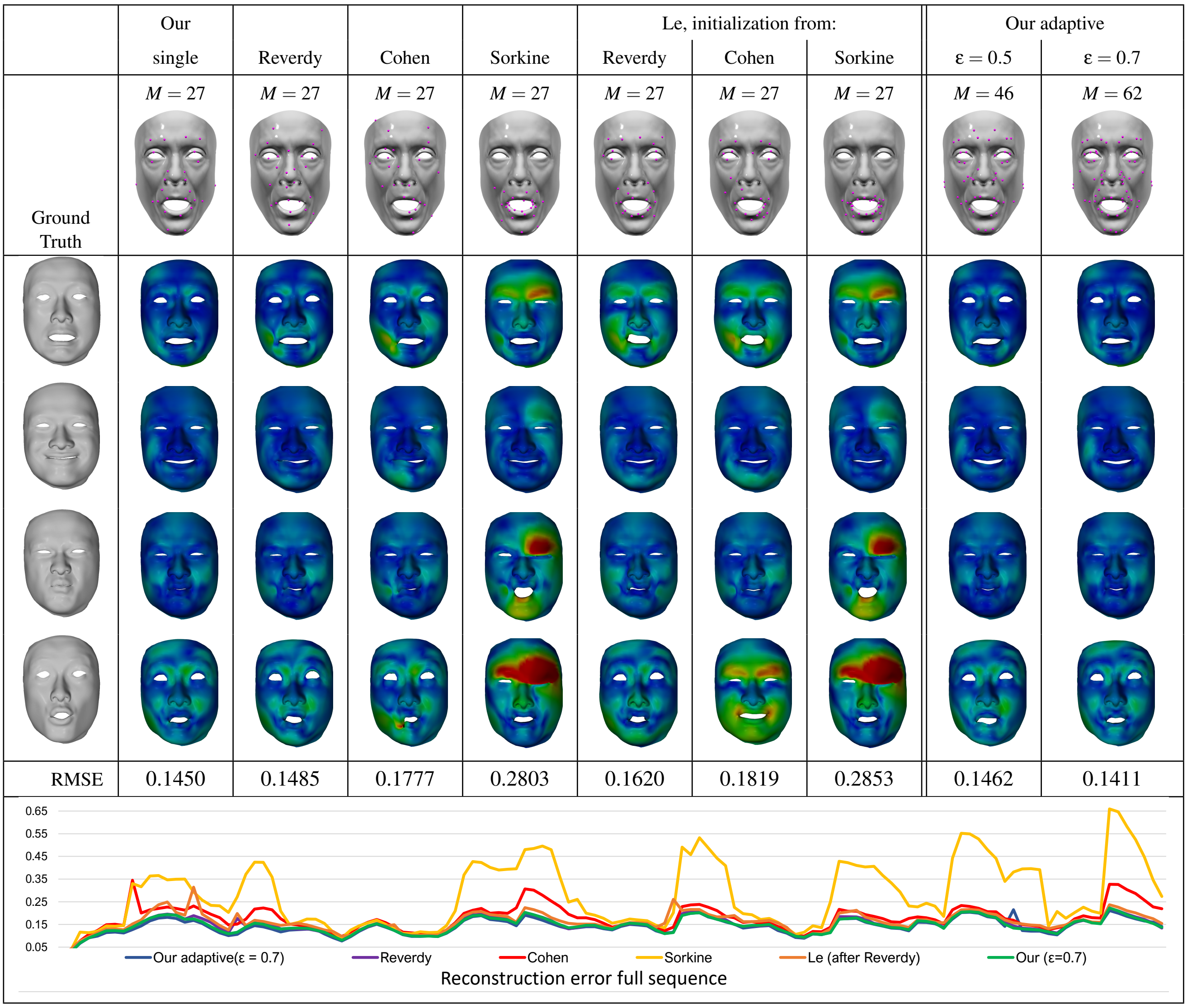
Sparse Animation Reconstruction Finally, we compare our layouts under real conditions with a VICON tracking system. A superset layout is constructed by considering different layouts by varying ε and the MPEG layout, which had the lowest number of blendshapes without a representative landmark. For practical reasons, the construction of the superset required to merge nearby landmarks to one or to shift their location slightly. For animation reconstruction, we follow closely the recent method of Ribera et al. [RZL*17]. This test is the most difficult, because it combines inaccuracies introduced by manual marker placement and retargeting issues between different characters. Fig. 10, 13 and the supplemental material show the results, where the superset layout serves as a baseline. Overall, the semantics of the animation were well preserved in all configurations, despite a significantly smaller number of landmarks. For moderate facial expressions, the difference between the results using the superset layout (baseline) and our layout is small to invisible. For exaggerated expressions, that are always challenging in retargeting scenarios, the visual differences become more noticeable, especially around the mouth, but the overall expression remains largely the same.

4.2. Minimalistic Layout for FACS

Our method computes minimalistic layouts for each blendshape model. At the same time we acknowledge the value of standardized landmark layouts, like MPEG-4 or MultiPIE. Considering the Facial Action Coding System [EF78, EHF81] as the most agreed semantic decomposition of facial expressions in computer science and psychology, we built a fully FACS compliant blendshape model, based on the FLAME/3D FACS dataset [LBB*17, CKH11]. For reference purposes, we computed symmetric landmark layouts with different settings for ε (supplemental material). Despite best efforts, a noticeable discrepancy existed between the layouts computed for the FACS model and the previously mentioned, FACS inspired, blendshape models (Fig. 2). Reasons for discrepancy could be missing posing accuracy, registration and blendshape decomposition artifacts or simply the difference between real FACS and idealized blendshapes(Fig. 14). To propose a more general layout for facial animation, we slightly modified the layout of the FACS model with $\varepsilon = 0.6$ (Fig. 2, Right and supplemental video).

5. Conclusion

In this paper, we proposed a novel algorithm for computing minimalistic facial landmark layouts specific to a blendshape model. Our computed layouts are more compact than empirically derived



	Our single	Reverdy	Cohen	Sorkine	Le, initialization from: Reverdy	Cohen	Sorkine	Our adaptive ε = 0.5	ε = 0.7
Ground Truth	$M = 27$	$M = 27$	$M = 27$	$M = 27$	$M = 27$	$M = 27$	$M = 27$	$M = 46$	$M = 62$
RMSE	0.1450	0.1485	0.1777	0.2803	0.1620	0.1819	0.2853	0.1462	0.1411

Figure 12: *Comparison against 4D data. Top row: Facial landmark layouts, consisting of $M = 27$, $M = 46$ or $M = 62$ landmarks in total, computed by respective methods for a male blendshape model from Epic's Metahuman (Fig. 1) with 157 blendshapes. Lower rows: The total RMS-error for the entire sequence, error maps for selected frames and error-distribution over the sequence after reconstructing the 4D sequence with a personalized blendshape model.*

layouts, have a small reconstruction error and create more plausible results than previous methods. Furthermore, we have shown that our method can derive layouts from FACS [EF78], which offers high potential to derive better standards for facial landmarks. The latter is currently the method of choice for facial animation. The novelty of our contribution is the insight that information loss is inevitable during the mapping to a low dimensional blendshape model paired with the careful exploration of how this affects the entire facial animation pipeline. Our analysis facilitated the formulation of the problem to find a minimal landmark layout as a linear integer optimization.

Comparing our results with existing facial landmarks or facial landmark layouts reveals that especially the chin, eyebrows, and parts of the nose are often over-sampled and the cheeks undersampled. The required total number of facial landmark could be smaller than existing empirical layouts indicate which offers a large potential to save resources when annotating large face datasets.

Direct blendshape manipulation methods [LAR*14] could also profit from optimal landmark layouts as this reduces the number of handles to control. Certainly, facial animation compression is not the main scope of our work, but it remains remarkable that the memory required to save the landmark positions of our layouts is

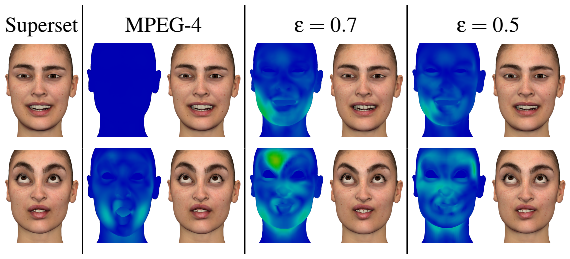


Figure 13: *Heat-maps showing the RMSE-error and the corresponding reconstructed frame. The superset serves as the baseline for comparison. The visible discrepancies between reconstructed frames are smooth deformations and thus remain barely noticeable on the textured character.*

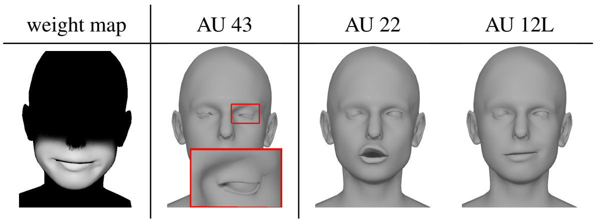


Figure 14: *Examples of: (left) a weight map for FACS decomposition, (middle) present registration artifacts in the FLAME/3D FACS dataset for Action Unit 43 which encodes a closed eye, (right) two successfully decomposed action units (AU22, AU12L).*

comparable or even smaller than the memory requirement to save the blendshape weights ($3M$ similar or less than K).

Finally, the approach of defining suitable landmarks to drive specific blendshapes can be exploited in cases of partial facial occlusion either due to hands, small objects, the presence of glasses, or VR headsets. If parts of the face are occluded, our metrics allow a differentiation between blendshapes that can be computed reliably from the visible face and those which should be estimated based on motion priors. In order to facilitate future work, a reference implementation of our method and references to the blendshape models will be published with the paper.

Acknowledgements

This research was partially funded by Science Foundation Ireland under the RADICal (Grant No. 19/FFP/6409), the ADAPT Centre for Digital Content Technology (13/RC/2106_P2) and Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy, EXC-2070—390732324—PhenoRob. Special thanks to Stephan Held and Donal Egan for fruitful discussions, Eisko for the VFX model and SideFX for software licences.

References

[AZCP13] Asthana A., Zafeiriou S., Cheng S., Pantic M.: Robust discriminative response map fitting with constrained local models. In *Proc. of Conf. on Computer Vision and Pattern Recognition* (2013), CVPR. 2

[BB14] Beeler T., Bradley D.: Rigid stabilization of facial expressions. *ACM Trans. Graph. 33*, 4 (July 2014). doi:10.1145/2601097.2601182. 3

[BBA*07] Bickel B., Botsch M., Angst R., Matusik W., Otaduy M., Pfister H., Gross M.: Multi-scale capture of facial geometry and motion. *ACM Trans. Graph. 26*, 3 (July 2007). doi:10.1145/1276377.1276419. 2

[BBB*14] Bermano A. H., Bradley D., Beeler T., Zund F., Nowrouzezahrai D., Baran I., Sorkine-Hornung O., Pfister H., Sumner R. W., Bickel B., Gross M.: Facial performance enhancement using dynamic shape space analysis. *ACM Trans. Graph. 33*, 2 (2014). doi:10.1145/2546276. 3

[BJKK11] Belhumeur P. N., Jacobs D. W., Kriegman D. J., Kumar N.: Localizing parts of faces using a consensus of exemplars. In *CVPR 2011* (2011), pp. 545–552. 2, 7

[BZLM18] Baltrusaitis T., Zadeh A., Lim Y. C., Morency L.: Openface 2.0: Facial behavior analysis toolkit. In *IEEE Conf. on Automatic Face & Gesture Recognition* (2018), FG, pp. 59–66. 2

[CET98] Cootes T., Edwards G., Taylor C.: Active appearance models. In *Proc. of the European Conference on Computer Vision* (1998), vol. 2 of *ECCV*, p. 484–498. 2

[CHZ14] Cao C., Hou Q., Zhou K.: Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Transactions on Graphics (TOG) 33*, 4 (2014), 43:1–43:10. 2

[CKH11] Cosker D., Krumhuber E., , Hilton A.: A facs valid 3d dynamic action unit database with applications to 3d dynamic morphable facial modeling. In *Int. Conf. on Computer Vision* (2011), pp. 2296–2303. 2, 9

[CSPF12] Chen X., Saparov A., Pang B., Funkhouser T.: Schelling points on 3d surface meshes. *ACM Trans. Graph. 31*, 4 (2012). doi:10.1145/2185520.2185525. 4

[CVW19] Chaudhuri B., Vesdapunt N., Wang B.: Joint face detection and facial motion retargeting for multiple faces. In *IEEE Conf. on Computer Vision and Pattern Recognition* (2019), CVPR, pp. 9719–9728. 2

[CXZ*02] Cohen I., Xiang Q. T., Zhou S., Sean X., Thomas Z., Huang T. S.: Feature selection using principal feature analysis, 2002. 2, 3, 4, 5, 6, 7, 8

[DYOY18] Dong X., Yan Y., Ouyang W., Yang Y.: Style aggregated network for facial landmark detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (June 2018), pp. 379–388. doi:10.1109/CVPR.2018.00047. 2

[EF78] Ekman P., Friesen W. V.: *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press, 1978. 3, 9, 10

[EHF81] Ekman P., Hager J. C., Friesen W. V.: The symmetry of emotional and deliberate facial actions. *Psychophysiology 18*, 2 (1981), 101–106. 9

[FHCP19] Fan Z., Hu X., Chen C., Peng S.: Boosting local shape matching for dense 3d face correspondence. In *IEEE Conf. on Computer Vision and Pattern Recognition* (2019), CVPR, pp. 10944–10954. 2

[GMC*10] Gross R., Matthews I., Cohn J., Kanade T., Baker S.: Multi-pie. *Image and Vision Computing 28*, 5 (2010), 807 – 813. Best of Automatic Face and Gesture Recognition 2008. doi:https://doi.org/10.1016/j.imavis.2009.08.002. 2, 7

[GPKZ19] Gecer B., Ploumpis S., Kotsia I., Zafeiriou S.: GANFIT: generative adversarial network fitting for high fidelity 3d face reconstruction. In *IEEE Conf. on Computer Vision and Pattern Recognition* (2019), CVPR, pp. 1155–1164. 2

[HLZC01] Hou X., Li S., Zhang H., Cheng Q.: Direct appearance models. In *Proc. of Conf. on Computer Vision and Pattern Recognition* (2001), vol. 1 of *CVPR*, p. pp. 828–833. 2

[Kar72] Karp R. M.: Reducibility among combinatorial problems. *Complexity of Computer Computations* (1972), 85–103. 5

[KMS11] Kholgade N., Matthews I., Sheikh Y.: Content retargeting using parameter-parallel facial layers. In *Proc. of Symposium on Computer Animation* (2011), SCA, pp. 195–204. doi:10.1145/2019406.2019433. 2

[KS14] Kazemi V., Sullivan J.: One millisecond face alignment with an ensemble of regression trees. In *Proc. of Conf. on Computer Vision and Pattern Recognition* (2014), CVPR, pp. 1867–1874. doi:10.1109/CVPR.2014.241. 2

[KS21] Kim J., Singh K.: Optimizing ui layouts for deformable face-rig manipulation. *ACM Trans. Graph. 40*, 4 (July 2021). doi:10.1145/3450626.3459842. 3

[KWRB11] Köstinger M., Wohlhart P., Roth P. M., Bischof H.: Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)* (2011), pp. 2144–2151. 2, 7

[LAR*14] Lewis J. P., Anjyo K., Rhee T., Zhang M., Pighin F. H., Deng Z.: Practice and theory of blendshape facial models. *Eurographics (State of the Art Reports) 1*, 8 (2014), 2. 3, 7, 10

[LBB*17] Li T., Bolkart T., Black M. J., Li H., Romero J.: Learning a model of facial shape and expression from 4d scans. *ACM Transactions on Graphics (TOG) 36*, 6 (Nov. 2017), 194:1–194:17. 2, 9

[LBL*12] Le V., Brandt J., Lin Z., Bourdev L., Huang T. S.: Interactive facial feature localization. In *Computer Vision – ECCV 2012* (2012), Fitzgibbon A., Lazebnik S., Perona P., Sato Y., Schmid C., (Eds.), Springer Berlin Heidelberg, pp. 679–692. 2

[Low04] Lowe D.: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision 60* (2004), 91–110. doi:10.1023/B:VISI.0000029664.99615.94. 4

[LSP08] Li H., Sumner R. W., Pauly M.: Global correspondence optimization for non-rigid registration of depth scans. In *Proceedings of the Symposium on Geometry Processing* (2008), SGP '08, Eurographics Association, p. 1421–1430. 8

[LZD13] Le B. H., Zhu M., Deng Z.: Marker optimization for facial motion acquisition and deformation. *IEEE Transactions on Visualization and Computer Graphics 19*, 11 (Nov 2013), 1859–1871. doi:10.1109/TVCG.2013.84. 2, 3, 4, 5, 6, 8, 9

[LZWM06] Liu G., Zhang J., Wang W., McMillan L.: Human motion estimation from a reduced marker set. In *Proc of ACM Symp. on Interactive 3D Graphics and Games* (2006), I3D, p. 35–42. 3, 5

[LZZL16] Liu F., Zeng D., Zhao Q., Liu X.: Joint face alignment and 3d face reconstruction. In *European Conference on Computer Vision* (2016), ECCV, pp. 545–560. 2

[Mar98] Martinez A. M.: The ar face database. *CVC Technical Report 24* (1998). URL: https://ci.nii.ac.jp/naid/10011462458/en/. 2, 7

[MB04] Matthews I., Baker S.: Active appearance models revisited. *International Journal of Computer Vision 60*, 2 (Nov 2004), 135–164. doi:10.1023/B:VISI.0000029666.37597.d3. 2

[MJC*08] Ma W.-C., Jones A., Chiang J.-Y., Hawkins T., Frederiksen S., Peers P., Vukovic M., Ouhyoung M., Debevec P.: Facial performance synthesis using deformation-driven polynomial displacement maps. In *ACM SIGGRAPH Asia 2008 Papers* (2008), SIGGRAPH Asia '08, pp. 121:1–121:10. doi:10.1145/1457515.1409074. 2

[MMK*99] Messer K., Matas J., Kittler J., Luettin J., Maitre G.: Xm2vtsdb: The extended m2vts database. In *Second international conference on audio and video-based biometric person authentication* (1999), vol. 964, pp. 965–966. 2, 7

[NJW01] Ng A. Y., Jordan M. I., Weiss Y.: On spectral clustering: Analysis and an algorithm. In *Proc. of Int. Conf. on Neural Information Processing Systems: Natural and Synthetic* (2001), NIPS, pp. 849–856. 3

[NVW*13] Neumann T., Varanasi K., Wenger S., Wacker M., Magnor M., Theobalt C.: Sparse localized deformation components. *ACM Trans. Graph. 32*, 6 (nov 2013). doi:10.1145/2508363.2508417. 3

[PJH16] Pharr M., Jakob W., Humphreys G.: *Physically Based Rendering: From Theory to Implementation*. Morgan Kaufmann, 2016. 4

[RGL15] Reverdy C., Gibet S., Larboulette C.: Optimal marker set for motion capture of dynamical facial expressions. In *Proc. of Motion in Games* (2015), MIG '15, p. 31–36. doi:10.1145/2822013.2822042. 3, 4, 6, 7, 8

[Rus04] Rusinkiewicz S.: Estimating curvatures and their derivatives on triangle meshes. In *Proceedings. 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004.* (2004), pp. 486–493. doi:10.1109/TDPVT.2004.1335277. 4

[RZL*17] Ribera R. B. i., Zell E., Lewis J. P., Noh J., Botsch M.: Facial retargeting with automatic range of motion alignment. *ACM Transactions on Graphics (TOG) 36*, 4 (July 2017), 154:1–154:12. 9

[SAT*16] Sagonas C., Antonakos E., Tzimiropoulos G., Zafeiriou S., Pantic M.: 300 faces in-the-wild challenge. *Image Vision Comput. 47*, C (Mar. 2016), 3–18. doi:10.1016/j.imavis.2016.01.002. 2

[SC04] Sorkine O., Cohen-Or D.: Least-squares meshes. In *Int. Conf. on Shape Modeling and Applications SMI* (2004), IEEE Computer Society, pp. 191–199. doi:10.1109/SMI.2004.1314506. 6, 8

[Sey16] Seymour M.: Put your (digital) game face on, 2016. URL: https://www.fxguide.com/featured/put-your-digital-game-face-on/. 2

[SLS*12] Seol Y., Lewis J., Seo J., Choi B., Anjyo K., Noh J.: Spacetime expression cloning for blendshapes. *ACM Transactions on Graphics (TOG) 31*, 2 (Apr. 2012), 14:1–14:12. 2

[SML16] Seol Y., Ma W.-C., Lewis J. P.: Creating an actor-specific facial rig from performance capture. In *Proceedings of the 2016 Symposium on Digital Production* (2016), DigiPro '16, pp. 13–17. 2

[SP04] Sumner R. W., Popović J.: Deformation transfer for triangle meshes. In *ACM Transactions on Graphics (TOG)* (2004), vol. 23, 3, ACM, pp. 399–405. 8

[SWR*18] Schröder M., Waltemate T., Röhlig T., Maycock J., Ritter H., Botsch M.: Design and evaluation of reduced marker layouts for hand motion capture. *Computer Animation and Virtual Worlds 29*, 6 (2018). 3, 5, 6

[TDlTM11] Tena J. R., De la Torre F., Matthews I.: Interactive region-based linear 3d face models. *ACM Trans. Graph. 30*, 4 (jul 2011). doi:10.1145/2010324.1964971. 3

[TGL*18] Tan Q., Gao L., Lai Y.-K., Yang J., Xia S.: Mesh-based autoencoders for localized deformation component analysis. In *Proc. of AAAI Conference on Artificial Intelligence and Innovative Applications* (2018), AAAI'18, AAAI Press. 3

[TKC01] Tian Y., Kanade T., Cohn J. F.: Recognizing action units for facial expression analysis. *IEEE Trans. Pattern Anal. Mach. Intell. 23*, 2 (2001), 97–115. 2

[TZS*16] Thies J., Zollhöfer M., Stamminger M., Theobalt C., Niessner M.: Face2face: Real-time face capture and reenactment of RGB videos. In *IEEE Conf. on Computer Vision and Pattern Recognition* (2016), CVPR, pp. 2387–2395. 2

[WCL*19] Wu W., Cao K., Li C., Qian C., Loy C. C.: Transgaga: Geometry-aware unsupervised image-to-image translation. In *Proc.of Computer Vision and Pattern Recognition (CVPR)* (June 2019). 3

[WGJ17] Wu Y., Gou C., Ji Q.: Simultaneous facial landmark detection, pose and deformation estimation under facial occlusion. In *IEEE Conf. on Computer Vision and Pattern Recognition* (2017), CVPR, pp. 5719–5728. 2

[Wil90] Williams L.: Performance-driven facial animation. In *Proc. of Conf. on Computer Graphics and Interactive Techniques* (1990), SIGGRAPH, pp. 235–242. doi:10.1145/97879.97906. 2

[WJZ13] Wheatland N., Jörg S., Zordan V.: Automatic hand-over animation using principle component analysis. In *Proc. of ACM Motion in Games* (2013), MIG, p. 175:197–175:202. 3

[WML21] Wang T.-C., Mallya A., Liu M.-Y.: One-shot free-view neural talking-head synthesis for video conferencing. In *Proc.of Computer Vision and Pattern Recognition (CVPR)* (June 2021), pp. 10039–10049. 3

[YLC*19] Yi H., Li C., Cao Q., Shen X., Li S., Wang G., Tai Y.: Mmface: A multi-metric regression network for unconstrained face reconstruction. In *IEEE Conf. on Computer Vision and Pattern Recognition* (2019), CVPR, pp. 7663–7672. 2

[ZDKZ19] Zhou Y., Deng J., Kotsia I., Zafeiriou S.: Dense 3d face decoding over 2500fps: Joint texture & shape convolutional mesh decoders. In *IEEE Conf. on Computer Vision and Pattern Recognition* (2019), CVPR, pp. 1097–1106. 2

[ZSCS04] Zhang L., Snavely N., Curless B., Seitz S. M.: Spacetime faces: High resolution capture for modeling and animation. *ACM Trans. Graph. 23*, 3 (aug 2004), 548–558. doi:10.1145/1015706.1015759. 8

[ZTG*18] Zollhöfer M., Thies J., Garrido P., Bradley D., Beeler T., Pérez P., Stamminger M., Niessner M., Theobalt C.: State of the art on monocular 3d face reconstruction, tracking, and applications. *Comput. Graph. Forum 37*, 2 (2018), 523–550. doi:10.1111/cgf.13382. 2