FaceDirector: Continuous Control of Facial Performance in Video

Charles Malleson[‡§*], Jean-Charles Bazin[‡*], Oliver Wang[‡], Derek Bradley[‡], Thabo Beeler[‡],
Adrian Hilton[§], Alexander Sorkine-Hornung[‡]
[‡]Disney Research Zurich
[§]Centre for Vision, Speech and Signal Processing, University of Surrey, UK

Abstract

We present a method to continuously blend between multiple facial performances of an actor, which can contain different facial expressions or emotional states. As an example, given sad and angry video takes of a scene, our method empowers a movie director to specify arbitrary weighted combinations and smooth transitions between the two takes in post-production. Our contributions include (1) a robust nonlinear audio-visual synchronization technique that exploits complementary properties of audio and visual cues to automatically determine robust, dense spatio-temporal correspondences between takes, and (2) a seamless facial blending approach that provides the director full control to interpolate timing, facial expression, and local appearance, in order to generate novel performances after filming. In contrast to most previous works, our approach operates entirely in image space, avoiding the need of 3D facial reconstruction. We demonstrate that our method can synthesize visually believable performances with applications in emotion transition, performance correction, and timing control.

1. Introduction

In film and television production, arguably one of the most important elements in achieving a believable and entertaining story is the performance of the actors. A key challenge lies in conveying believable emotions with appropriate facial expressions, speed and timing. As a consequence, scenes are often shot and re-shot over and over as multiple takes until the director is satisfied, often requiring considerable amounts of time and cost. For example, the opening scene of the movie "The Social Network" required 99 takes, "Gone Girl" required an average of 50 takes per scene and one scene in "The Shining" required 127 takes[1]. To help alleviate this problem, we propose a continuous facial performance interpolation approach, enabling the director to synthesize a wide variety of novel performances after capture, e.g., in post-production, from a much sparser set of takes, as illustrated in Fig. 1.

Figure 1: Given a pair of facial performance videos (here an angry and a happy version of a monolog denoted $\boldsymbol{v}_0$ and $\boldsymbol{v}_1$, respectively), our method automatically computes a nonlinear temporal synchronization based on facial expression and audio cues, illustrated by the blue synchronization path. We can then seamlessly blend between the performances both with respect to time and expresssion and synthesize a novel interpolated performance (see film strip) with intuitive artistic control (see green blending curve).

A central challenge of interpolating performances is video synchronization. Synchronization using simple constant time offsets or uniform temporal scaling of the input videos is not feasible because of the complex nonlinear local variations in timing and speed during facial performances. Difference of head pose, emotion, expression intensity, as well as pitch, accentuation and potentially even wording of the speech are just a few of the many difficulties. We present an automatic, joint audio-visual synchronization approach that first analyzes both facial expression and audio cues and then robustly determines a dense set of frame correspondences between takes using a graph-based framework. To the best of our knowledge, our work is the first to combine audio and facial features for achieving an optimal nonlinear temporal alignment of performance videos.

A PCA-based facial landmark normalization is used to cope with large variations of the landmarks with different facial expressions and emotions. Furthermore, we show that an important aspect for a successful automatic synchronization using cost matrices is the removal of ambiguous, self-similar parts, which are unavoidable when using local descriptors on highly redundant data such as facial perfor-

[*] denotes joint first authorship with equal contribution
[1] http://en.wikipedia.org/wiki/Take

mances. Once the videos are synchronized, we propose a nonlinear spatio-temporal performance blending method that blends across timing, facial expression and local appearance changes. Our method smoothly blends the videos using a locally time varying parameterization of a synchronizing path computed on performance cost matrices, and blends both expression and appearance using facial landmarks, optical flow and compositing for a seamless combination and interpolation of the input videos (see Fig. 1). Our approach is passive, operating directly on 2D input video footage from a single camera without the need for additional hardware or 3D facial reconstruction.

2. Related work

Our work is related to facial performance capture and manipulation, and video synchronization and blending. We now discuss the most related methods in these areas.

Facial performance capture and manipulation. One common approach is to create a digital avatar of the actor [1], which can then be animated as desired. Creating a believable digital double of an actor is a very challenging and time consuming task involving high-resolution facial capture [26, 7, 17, 2, 21] to construct the rig. Once a high-resolution actor-specific rig has been created, it can be driven [27, 30] from different input modalities, such as binocular video [38, 5], monocular video [16, 32, 10], depth sensor [6, 23], or existing detailed animated face models [42]. Given a sufficiently detailed animated 3D rig or database of 3D facial expressions, it is also possible to use a video of an actor to drive visual speech synthesis [36] or to replace parts of a facial performance in video, e.g., for language dubbing [15]. However, in general, creating a digital double is only practical for large-budget productions and even there is typically restricted to a few hero characters.

To avoid the need for a high-resolution actor-specific rig, some works propose to leverage a low-resolution generic face rig to manipulate acquired video footage. Kuster et al. [22] modify the pose of the face to synthesize eye-contact in video chats via an RGBD sensor. Dale et al. [11] employ a 3D morphable face model [39] in order to replace facial performances in video footage of the same or from a different person. Their use-case differs from ours in that they completely replace one facial performance by another, while we wish to smoothly blend between the performances both spatially and temporally to synthesize a novel performance. Closer in spirit to ours is the work of Yang et al. [43] which can exaggerate or attenuate facial expressions in an input video via a 3D face model. In contrast, our method allows creative interpolation of facial performances without the requirement of 3D facial reconstruction.

A last alternative is to operate entirely in image space and to directly manipulate the captured footage, as we do in this work, thus avoiding the need for a 3D face prior altogether. Bregler et al. [8] were the first to synthesize a video sequence according to a new audio track by computing a re-ordered mouth sequence from training data. Ezzat et al. [13] extend this further and use a trained morphable model to synthesize novel speech sequences. Kemelmacher-Shlizerman et al. [20] aim at puppeteering a person by aligning input images to a large image database of the same person, e.g., leveraging community photo collections. A challenge there is to avoid discontinuities in the output, due to the heterogeneity and limited sampling of these collections. To overcome this, Garrido et al. [14] capture the input samples themselves, and warp the retrieved images in order to replace the face. This differs from our application, where the goal is to continuously blend between two or more performances both spatially and temporally to produce a novel performance. Also related is the work of Berthouzoz et al. [4], which aims to place cuts and create seamless transitions in interview videos. Their method is well suited to remove undesired parts of an interview or potentially reshuffle the sentences but not to modify or interpolate facial performances.

Video-based synchronization. Temporal synchronization and spatial alignment of videos is a crucial step for blending and interpolation. Various techniques exist for computing such alignments on general video sequences [31, 12, 40]. However, these methods estimate correspondences with general purpose appearance-based descriptors, and work best with camera ego-motion and large-scale scene changes. While these assumptions are reasonable for general video, for facial performances it is essential to integrate higher level knowledge into the process to be able to distinguish, e.g., between the global head motion and the subtle changes of facial expressions.

For these reasons, some works have focused on solutions specifically designed for synchronizing human performances. For example, Hsu et al. [19] present a method for style translation and motion retargeting based on motion capture data. Zhou and De la Torre [44, 45] present methods based on time warping for aligning motion of multiple human subjects performing similar actions. However, these application specific approaches are not straightforward to extend to accurate, spatio-temporal alignment as required for blending between facial performances. Closest to our synchronization method are the facial alignment techniques of Dale et al. [11] and Yang et al. [43] discussed above. Dale et al. [11] apply dynamic time warping on the velocities of the mouth vertices. While this is appropriate for spoken videos, it cannot handle general (e.g., silent) facial performances. Yang et al. [43] also use dynamic time warping, but on the expression coefficients of their morphable face model. This approach works well for very expressive facial motion, but is limited with respect to subtle fa-

cial changes in expression. In contrast to these approaches, our method uses facial landmarks extracted at several locations of the face and considers the spatial distribution of each landmark for normalization. In addition, our method also utilizes synchronization cues from audio data, which we found to be crucial for achieving robust temporal alignment where facial visual information alone is ambiguous. We build our synchronization technique upon the work of Wang et al. [40] and show that, by modifying their basic framework with a redesigned feature space and a novel approach for removing self-similarities from cost matrices, audio and visual information can complement each other and lead to highly robust, automatic and effective temporal alignment of facial performances.

Image and video blending. Once an alignment between two frames of a video sequence has been found, various image-based techniques for compositing [29, 35] or morphing [3, 24, 25] exist. However, many of these methods require manual correspondences, manual refinement, or alignment of regions to be blended. Similar to the above discussion on video synchronization, they are designed for general purpose blending between arbitrary images or videos. Instead we aim for a method with automatic correspondence computation that can robustly obtain a believable and realistic facial expression without any ghosting or other alignment artifacts.

3. Method overview

Given a pair of monocular video takes, we wish to create a novel performance by smoothly blending them both in space and time. Our work focuses on the classic medium and close-up frontal head shots (see Fig. 1), and a relatively fixed filming setup. Such head shots are particularly challenging because the full attention of the viewer is directed to the actor's face. The algorithm consists of two main steps: nonlinear synchronization of the input takes to establish proper temporal correspondences (Sec. 4), and spatio-temporal seamless blending of the synchronized takes (Sec. 5). When blending, we use the scene background and global head motion provided in one of the input videos (the first one, without loss of generality) into which we composite the interpolated interior of the face.

For the synchronization step we extend the method of Wang et al. [40], which temporally aligns videos based on a cost matrix that encodes the alignment quality between each pair of video frames. The nonlinear synchronization is then given as the minimum average cost path through the cost matrix. In this work we show that a robust synchronization of performance takes can be achieved by tailoring the cost matrices to facial performances by employing distinct features such as facial landmarks and audio cues, and by removing ambiguous information from the cost matrices.

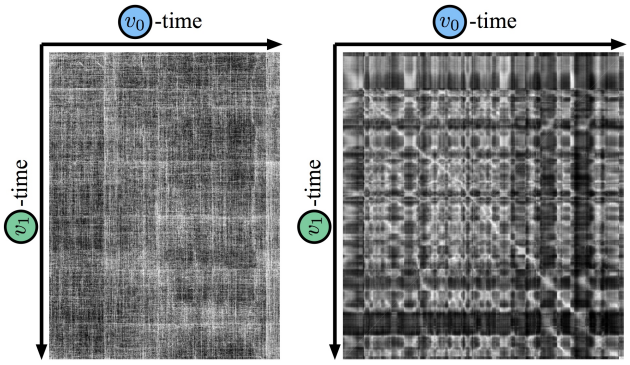


Figure 2: The cost matrix on the left has been computed from two facial performance input videos with general purpose appearance descriptors [40], and contains no obvious path-like structures that could be used for temporal synchronization. On the right is the corresponding cost matrix computed with our approach, with a rather clear low cost path along the diagonal (bright colors correspond to low cost, dark to high). The different aspect ratios of the matrices stem from our adaptive matrix collapsing approach.

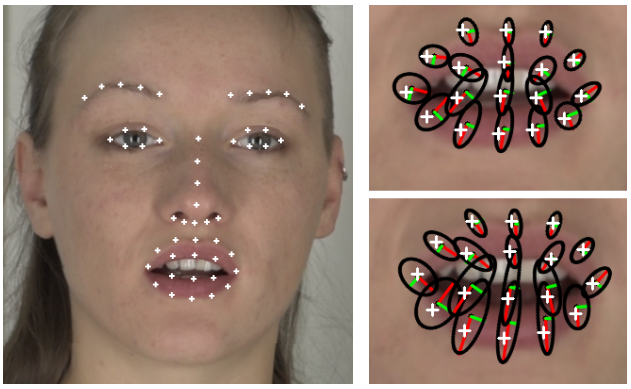
Performance blending is then achieved by traversing the synchronization path through the input videos, and computing weighted combinations of each expression based on any user-specified blending function $\alpha(t)$.

4. Performance synchronization

Let $\boldsymbol{v}_0$ and $\boldsymbol{v}_1$ be two input video takes, and $\boldsymbol{v}_i(j)$ be the j-th frame of video $\boldsymbol{v}_i$. A temporal synchronization is then defined as a mapping $\boldsymbol{p} : \mathbb{R} \rightarrow \mathbb{R}^2$, where $\boldsymbol{p}(t) = (p_0(t), p_1(t))$ associates a global time t with two corresponding frames $\boldsymbol{v}_0(p_0(t))$ and $\boldsymbol{v}_1(p_1(t))$. To estimate the mapping $\boldsymbol{p}$, we extend the path computation of Wang et al. [40]. Their general appearance-based features are not applicable in our setting since the appearance variation between frames of facial performances is too subtle (see Fig. 2). We therefore introduce domain-specific features based on normalized facial landmarks and audio cues, which allow us to robustly synchronize facial performances.

4.1. Feature extraction and processing

Facial landmarks. We use the IntraFace tracker [41] to obtain a set of 2D facial landmark features in all frames of the input videos (see Fig. 3a). To reduce noise in the landmark positions, we apply a bilateral filter [37] to each landmark (we used $\sigma_{time} = 5$ frames and $\sigma_{space} = 5\%$ of the pixel distance between the eye corner landmarks). We denote by $\mathbf{f}_0^i(j)$ the image coordinates of the i-th filtered landmark in the video frame $\boldsymbol{v}_0(j)$, and by $\mathbf{f}_1^i(k)$ the corresponding landmark in $\boldsymbol{v}_1(k)$.



(a) (b)

Figure 3: (a): input facial landmarks (b): per-landmark PCA local coordinate systems for a temporally corresponding frame pair from two takes with different emotions (top neutral, bottom sad). Note the performance-dependent orientations and scales of the PCA local coordinate systems.

Head pose estimation. We estimate head pose using a subset of landmarks that are relatively invariant to expression change, namely the bottom of the nose and the corners of the eyes. For all frames in $\boldsymbol{v}_0$ and $\boldsymbol{v}_1$, a 2D rigid transformation is computed [34] with respect to a reference frame, chosen arbitrarily as $\boldsymbol{v}_0(0)$. Finally, for each video, we use the estimated per-frame transformations to register all the facial landmarks to the pose of the reference frame. We denote by $\hat{\mathbf{f}}_0^i(j)$ and $\hat{\mathbf{f}}_1^i(k)$ the registered positions of the landmarks $\mathbf{f}_0^i(j)$ and $\mathbf{f}_1^i(k)$.

Normalized landmarks. The landmarks are used as constraints in the video synchronization by computing a pair-wise frame alignment cost that represents the similarity of the respective facial expressions. A simple way to measure the difference $d_L(j,k)$ between the landmarks of $\boldsymbol{v}_0(j)$ and $\boldsymbol{v}_1(k)$ is the sum of their squared Euclidean distances:

$$d_L(j,k) = \sum_i ||\hat{\mathbf{f}}_0^i(j) - \hat{\mathbf{f}}_1^i(k)||^2. \quad (1)$$

However, we observed that for the same scene performed with different emotions across takes, the range of facial articulation may vary substantially. To account for this we normalize each landmark as follows. For each video, Principal Component Analysis (PCA) is performed on each facial feature over all the frames. This gives a local origin, orientation and scale to each feature as shown in Fig. 3b. The PCA-normalized distance is then computed as:

$$\hat{d}_L(j,k) = \sum_i ||\hat{\mathbf{f}}_0^i(j) - \rho(\hat{\mathbf{f}}_1^i(k))||^2, \quad (2)$$

where ρ is the linear operation that aligns the origin, orientation and scales of $\hat{\mathbf{f}}_1^i$ with those of $\hat{\mathbf{f}}_0^i$ as determined by the PCA. Comparing the features in their local spaces yields distance estimates with improved invariance to scale and orientation changes, and hence leads to a more robust and informative measure for comparing facial performances.

We then apply a common decaying function to convert the distance $\hat{d}_L(j,k)$ into a similarity measure, i.e. small distance gets a high similarity:

$$s_L(j,k) = \exp(-\lambda \cdot \hat{d}_L(j,k))^\beta, \quad (3)$$

where λ and β are respectively set to 0.005 and 2 in all our experiments. The value $s_L(j,k)$ lies between 0 (dissimilar) and 1 (similar).

We experimented with alternative measures such as landmark velocities (as also employed by Dale et al. [11]), but found that for our application scenario, the landmark derivative information is too noisy and so this approach results in less reliable similarity measures.

Audio features. In addition to the synchronization cues extracted from visual information, we also use cues from the associated audio tracks, which is essential when the visual information of the face alone is ambiguous. To extract information from the audio data, we employ Mel-Frequency Cepstral Coefficients (MFCCs) [28], which are commonly employed descriptors in audio analysis, description and retrieval. We compare the MFCCs extracted over the duration of two frames $\boldsymbol{v}_0(j)$ and $\boldsymbol{v}_1(k)$ by computing their Euclidean distance, and denote this audio distance $d_A(j,k)$. Analogous to the case of facial landmarks, we convert the distance $d_A(j,k)$ into a similarity:

$$s_A(j,k) = \exp(-\lambda \cdot d_A(j,k))^\beta, \quad (4)$$

where λ and β have the same values as in Eq. 3. Again, these values are defined in the range $[0,1]$.

Note that audio similarity alone is not powerful enough for facial performance alignment because (1) the audio can vary greatly across takes due to different intonations and choice of wording, and (2) audio alone cannot handle purely visual performances (e.g., silent changes in facial expression). For this reason, we propose to combine both visual and audio information to robustly synchronize facial performances, as described in the next section.

4.2. Local cost matrix collapsing

The temporal synchronization of Wang et al. [40] based on minimum average cost path computation requires a cost matrix computed from the pair-wise frame similarities. A straightforward way to build such a matrix from our landmark and audio similarities $s_\star(j,k), \star \in \{L, A\}$ would be to convert them into cost matrix entries $c_\star(j,k)$ as follows:

$$c_\star(j,k) = \left(1 - \frac{s_\star(j,k)}{\max(s_\star)}\right)^\gamma, \quad (5)$$

where γ is a user-defined parameter, and then create a final cost matrix as a weighted combination

$$C = (1 - w)C_L + wC_A, \tag{6}$$

where w is a relative weight. A fundamental problem with this simple approach is illustrated in Fig. 4. Facial performances exhibit a high degree of self-similarity, both in terms of visual landmarks as well as auditory cues. As a result, the corresponding cost matrices contain large blocks of entries with considerable ambiguities. For example, in cases where the actor holds still, the landmarks do not contain any valuable information. Similarly, in parts of a performance where the actor remains quiet, the audio features are not informative. A weighted combination of both landmark and audio costs improves the situation, as both feature types often complement each other, but does not fully resolve these problems (Fig. 4a). The path computation may find an alignment that is wrongly biased by these self-similar blocks and misses parts of the correct synchronization (see Fig. 4c).

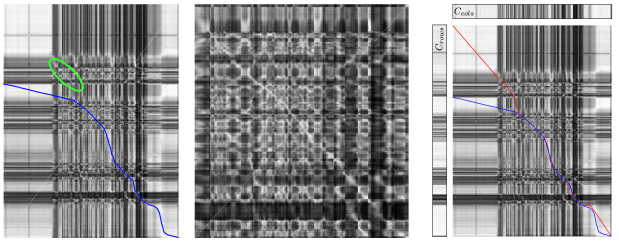
As a remedy, we propose to locally collapse uninformative rows and columns in the cost matrix, which significantly reduces the influence of these ambiguities and emphasizes cost matrix regions with a distinct signal. We therefore compute per-row sums of the cost

$$C_{rows}(j) = \sum_i C(j, i), \tag{7}$$

and remove row j if its sum is smaller than a conservative threshold τ, $C_{rows}(j) < \tau$. The same procedure is applied to the columns by computing $C_{cols}(j)$. The sums for the combined matrix in Fig. 4a are visualized in Fig. 4c. The result is a collapsed cost matrix $\tilde{C}$ as shown in Fig. 4b. Compared to the original cost matrix C, ambiguous regions that potentially deteriorate the path computation have been removed, leaving a cost matrix with a sufficiently distinct low cost path. Note that the removed rows and columns with large self-similar blocks cannot contain an actual signal relevant for synchronization, since the respective frames are inherently ambiguous.

After computing the path on $\tilde{C}$, we undo the collapsing and linearly interpolate the missing path fragments (red path in Fig. 4c). Due to the linear path extension in self-similar regions, it is ensured that both videos are played as close as possible to their original speed. The path maps a global time to the frames of the input videos $\boldsymbol{v}_0$ and $\boldsymbol{v}_1$, and the videos can then be temporally aligned with respect to a global time t: $\boldsymbol{p}(t) = (p_0(t), p_1(t))$ builds the temporal correspondences $\boldsymbol{v}_0(p_0(t))$ and $\boldsymbol{v}_1(p_1(t))$.

In all our experiments we used $\gamma = 2$ and $w = 0.5$. For τ it was sufficient to pick a conservative threshold $\tau = c_{min} + p(c_{max} - c_{min})$ with $p = 0.1$, where c_{min} and c_{max} respectively represent the min and max value in C_{rows} and C_{cols}, so that only highly self-similar regions are removed.



(a) combined audio and landmark cost C and computed path (b) our collapsed cost $\tilde{C}$ exhibiting a distinct low cost path (c) per-row/-column cost and our sync. path in C (red)

Figure 4: Even when combined, audio and landmark costs (a) contain self-similar low cost blocks that impede the computation of a correct path (in blue), leading to a wrong synchronization that misses a part of a dialog line (bright diagonal in highlighted area). In our collapsed cost $\tilde{C}$ (b) self-similarities without a reliable path signal are removed, leading to a more distinct path (bright diagonal), and hence a more robust path computation (in red) and a correct synchronization (c). The axes (v_0 and v_1) of these cost matrices correspond to those in Fig. 2.

5. Spatio-temporal performance blending

After the input videos are temporally synchronized, we compute a spatio-temporal blend between the takes. In order to accomplish this, we need to blend in multiple dimensions including timing, facial expression (shape), and local appearance. Creative control of the blending is achieved by using a continuous time-varying parameter $\alpha(t) \in [0, 1]$, where $\alpha(t) = 0$ (resp. $\alpha(t) = 1$) corresponds to the timing and appearance of $\boldsymbol{v}_0$ (resp. $\boldsymbol{v}_1$), and $0 < \alpha(t) < 1$ results in a visual blend between the two input performances. The function $\alpha(t)$ can be any interpolating function that the user desires, including nonlinear and non-monotonic interpolations, as we will show in Sec. 6. Note that, for the case of blending between takes of different emotions, our goal is to interpolate the *visual* appearance (and timing) and not necessarily to interpolate the actual emotions, e.g., $0.5 \times happy + 0.5 \times sad \neq neutral$. We retain the head pose and background from $\boldsymbol{v}_0$ rather than also blending the rest of the video frame, which may contain arbitrary scene elements and hence is a challenging problem in itself [24].

5.1. Temporal blending

In order to explain our temporal blending, consider the parameterization of our path $\boldsymbol{p}$ (see Fig. 6). The path $\boldsymbol{p}(t)$ gives us a pair of frames in correspondence at time t. However, we are free to arbitrarily navigate along t by choosing a particular parameterization of the path, i.e., by controlling the step size for t, we can advance either video at a desired rate. For example, taking unit-length steps along the axis of $\boldsymbol{v_0}$ would correspond to playing $\boldsymbol{v_0}$ at its original

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

Figure 5: Spatial blending process. **(a)** Input image pair from $\boldsymbol{v}_0$ (top) and $\boldsymbol{v}_1$ (bottom). **(b)** Closeups of the different facial expressions. **(c)** Pose normalized facial landmarks for $\boldsymbol{v}_0$ (blue) and $\boldsymbol{v}_1$ (green) overlaid on pose normalized images with mask. **(d)** Optical flow using predictions from landmarks. **(e)** Warped input images with $\alpha = 0.5$. **(f)** Final composite with visually interpolated expression.

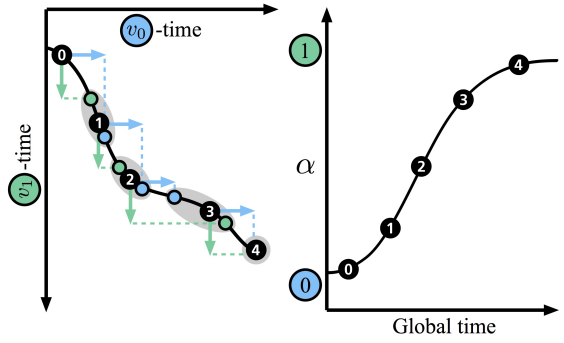


Figure 6: Time remapping of the synchronization path (left) based on a time varying blend curve for creative control (right). For both videos, unit length steps are computed along the path (illustrated by the blue and green arrows and circles, respectively), between which a blended time is computed according to $\alpha(t)$ (black circles).

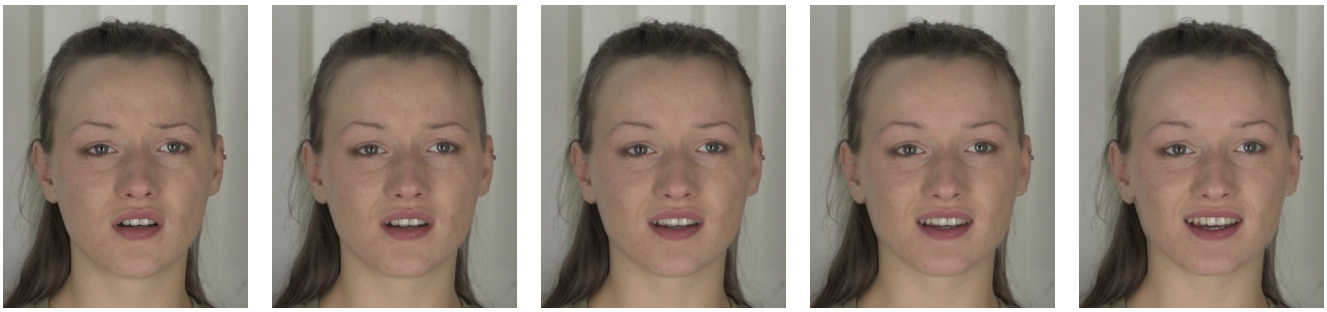
speed, while temporally remapping the video $\boldsymbol{v_1}$ to match the speed of $\boldsymbol{v_0}$. Our goal is to smoothly interpolate between the different speeds of $\boldsymbol{v_0}$ and $\boldsymbol{v_1}$ according to $\alpha(t)$, so that the timing of the output performance seamlessly blends between the two. To achieve this temporal blending, we determine the arc-length increment required for unit time steps in both input videos independently. We then compute an intermediate time point along the path that is located between these two points as determined by $\alpha(t)$ (see black circles in Fig. 6). The step size is therefore locally varying throughout the performance. Each intermediate time point provides a pair of corresponding frames in $\boldsymbol{v_0}$ and $\boldsymbol{v_1}$, and the collection of these frames constitutes the temporal blending of the performance videos.

5.2. Spatial blending

After the synchronized pair of frames at a particular time t has been found, we aim to generate a visually plausible interpolated performance frame according to $\alpha(t)$. An overview of the main steps is shown in Fig. 5. First, an interpolated frame is computed from the two synchronized input frames. Then, the resulting synthesized facial expression is composited back into $\boldsymbol{v_0}$. In the following, we explain these two steps in detail.

To create the spatially interpolated frame from $\boldsymbol{v_0}$ and $\boldsymbol{v_1}$, we utilize a modified optical flow-based warping [9]. First, we spatially align the faces from both input frames using the head poses estimated in Sec. 4 (see Fig. 5b). Large non-rigid displacements, e.g., large variation between the facial expressions in both input frames, are usually problematic for variational flow techniques. This problem can be alleviated by using the positions of the facial landmarks as a flow prior. An important detail here is to use the landmarks as soft rather than hard positional constraints in order to compensate for localization errors and noise in the landmark positions. An example of the aligned landmarks and a corresponding flow field are shown in Fig. 5c and 5d. The computed flow is then used to warp the two input frames with fractions $\alpha(t)$ and $1 - \alpha(t)$, respectively (see Fig. 5e).

There are various options to blend the two warped frames and composite them back into $\boldsymbol{v_0}$. To ensure robust and simple computation for fast visual feedback we found that a simple mask-based approach works well. In the first frame of the sequence, we build a color model of the face using the pixels inside the convex hull of the facial landmarks. The mask is then computed by detecting the face pixels in agreement with the color model and simple refining by morphological operators (see mask in Fig. 5c). An additive alpha-blend is performed between the warped source frames, and then the blended face is seamlessly composited back into $\boldsymbol{v_0}$ using the same alpha-blending approach (see Fig. 5f). We found that this simple method was sufficiently accurate for our application of facial performance blending between videos of the same actor, as shown in the experiments.



$\alpha = 0$ $\alpha = 0.25$ $\alpha = 0.5$ $\alpha = 0.75$ $\alpha = 1$

Figure 7: Final compositing results for interpolating a pair of matching frames from a 'sad' take (left) to a 'happy' take (right), with corresponding α values.

We also experimented with alternative, more sophisticated image blending techniques such as Poisson image editing [29]. However, we found that, for our particular application with facial performance videos, our solution produces more robust and higher quality results, mostly because integration-based approaches can be susceptible to color bleeding, flickering, and other temporal artifacts along the seam of the masks. Some compositing results of our method with varying values for α are shown in Fig. 7.

5.3. Audio rendering

To generate the output audio, the source audio signals need to be time-warped as well using the time remapping computed in Sec. 5.1. Naive time-warping by simple resampling of the audio signal would not preserve the pitch. While our work focuses on video processing aspects, we implemented a conventional pitch-preserving time-scale technique (WSOLA) [18], which produces an acceptable retimed audio preview by concatenating short audio segments from the appropriate points in the input. We expect that higher quality audio results can be achieved by employing more sophisticated audio processing algorithms [33] or professional audio retiming software.

6. Results

In the following we discuss relevant information about data capture and implementation, and present various results and applications of our method.

Input data capture. We provided our actors with a selection of dialog lines that were designed in a way that they could be performed with different underlying emotions. Each subject then performed the same line multiple times and each time tried to convey a different emotion such as happiness, sadness, excitement, anger, fear, etc. The videos were acquired with standard compact cameras at full HD resolution and 25fps.

Implementation details. We processed the videos on a desktop computer equipped with an Intel i7 3.2Ghz and 16GB RAM, on a C++ unoptimized, single-core implementation. The execution time for a typical 15-second clip recorded at full HD is as follows. During preprocessing, we extract the facial landmarks, which takes about 0.2s per frame, and the audio descriptors, which takes less than 1s per full video. Given these features, for a pair of videos, the path computation itself takes about 0.1s, and the flow and mask can all be precomputed (about 1.2s and 2s per frame, respectively). With our current implementation, results at the full HD resolution can be generated at interactive rates, with 90ms computation time per frame. Given the input videos, the synchronization runs in a fully automatic manner, and the user can generate novel versions of the performances with arbitrary creative control by simply interactively manipulating the $\alpha()$ blending curve.

Applications. Fig. 8 shows a representative result for temporal synchronization using our proposed approach. The first two rows show uniformly sampled frames from two input videos $\boldsymbol{v}_0$ and $\boldsymbol{v}_1$. In both videos, the subject performed the same dialog line with slightly different timing and varying facial expressions. The third row shows $\boldsymbol{v}_1$ after temporal alignment to $\boldsymbol{v}_0$, where the facial features like mouth shape are now synchronized.

Final interpolation results between input takes of an actor performing the same dialog line with different emotions are shown in Fig. 9. All images are composited frames taken from transition phases between the two input takes.

We kindly invite the readers to refer to our project webpage[2] for the full video results, as well as a user study, our dataset of facial performances and several additional results on video synchronization, emotion transition, takes acquired with hand-held cameras, blending with/without synchronization, acting directives, and generation of numerous performances from a sparse set of input takes.

[2] http://www.disneyresearch.com/publication/facedirector

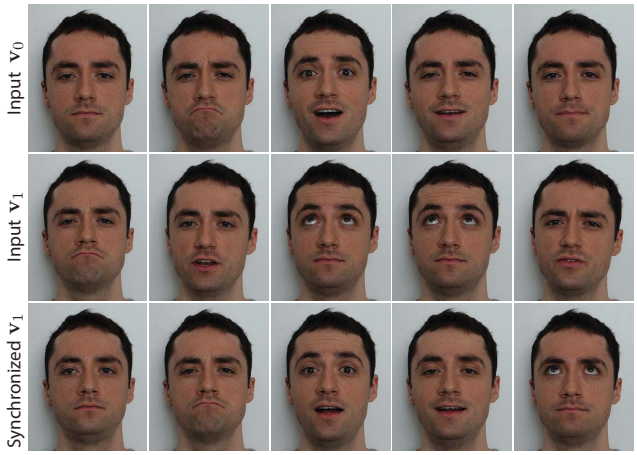


Figure 8: Top and middle rows show the two input sequences $\boldsymbol{v}_0$ and $\boldsymbol{v}_1$, respectively. Note how different the facial expressions are in the two sequences due to the synchronization mismatch. Bottom row: our method successfully synchronizes $\boldsymbol{v}_1$ with respect to $\boldsymbol{v}_0$. Please see the supplemental video for the complete result.

Figure 9: Interpolation results for different actresses with α transitioning from 0 to 1. The top row shows a transition from neutral to scared and the bottom row from sad to angry.

Limitations and future work. While working solely in the 2D image domain is beneficial in many aspects and often sufficient in our context, our method is limited with respect to large out of plane rotations, dynamic lighting, or large expression differences (see Fig. 10), some of which could be alleviated by using 3D trackers [10, 32]. We also do not explicitly handle glasses or hair that covers part of the face, a limitation common to existing works on video based face manipulation (e.g., [11, 43, 22]).

7. Conclusion

We have presented a new approach to continuously blend between videos of facial performances. Our key contributions are a robust and automatic approach to temporally and spatially align different takes, and a computationally simple but effective image blending approach. Experiments show that our approach can synthesize interpolated, visually plausible novel versions of the performances.

We believe that techniques for creative, interactive control over facial performance videos will gain increasing importance in both research and the industry, providing a wide range of interesting opportunities for followup work, e.g., on more complex subject motion and novel artistic effects.

Figure 10: Failure cases. Left, middle: input frames. Right: blended result with artifacts. Top row: head registration issue, where the actress directly faces the camera in one take and with the eye line off the camera in another take. The head registration with 2D landmarks did not handle this out of plane rotation properly. Bottom row: blending issue, where significantly different mouth shapes are not properly handled by the optical flow based registration.

Acknowledgements

We are very grateful to Prof. Anton Rey, Daniel Löpfe (camera operator), Svenja Koch, Johanna Köster, Marie Sophie Schmidt, Anna Rebecca Sehls, and Julian-Nico Tzschentke (actors) from the Zurich University of the Arts (ZHdK) for their invaluable help and great performances. Furthermore, we thank Max Grosse and Felix Klose for helping with the data capture, and Olga Sorkine-Hornung for insightful discussions on point registration.

References

[1] O. Alexander, M. Rogers, W. Lambeth, J. Chiang, W. Ma, C. Wang, and P. E. Debevec. The digital Emily project: Achieving a photorealistic digital actor. *IEEE Computer Graphics and Applications*, 2010. 2

[2] T. Beeler, F. Hahn, D. Bradley, B. Bickel, P. A. Beardsley, C. Gotsman, R. W. Sumner, and M. Gross. High-quality passive facial performance capture using anchor frames. *TOG (SIGGRAPH)*, 2011. 2

[3] T. Beier and S. Neely. Feature-based image metamorphosis. In *SIGGRAPH*, 1992. 3

[4] F. Berthouzoz, W. Li, and M. Agrawala. Tools for placing cuts and transitions in interview video. *TOG (SIGGRAPH)*, 2012. 2

[5] K. S. Bhat, R. Goldenthal, Y. Ye, R. Mallet, and M. Koperwas. High fidelity facial animation capture and retargeting with contours. In *SCA*, 2013. 2

[6] S. Bouaziz, Y. Wang, and M. Pauly. Online modeling for realtime facial animation. *TOG (SIGGRAPH)*, 2013. 2

[7] D. Bradley, W. Heidrich, T. Popa, and A. Sheffer. High resolution passive facial performance capture. *TOG (SIGGRAPH)*, 2010. 2

[8] C. Bregler, M. Covell, and M. Slaney. Video rewrite: driving visual speech with audio. In *SIGGRAPH*, 1997. 2

[9] T. Brox and J. Malik. Large displacement optical flow: Descriptor matching in variational motion estimation. *TPAMI*, 2011. 6

[10] C. Cao, Q. Hou, and K. Zhou. Displaced dynamic expression regression for real-time facial tracking and animation. *TOG (SIGGRAPH)*, 2014. 2, 8

[11] K. Dale, K. Sunkavalli, M. K. Johnson, D. Vlasic, W. Matusik, and H. Pfister. Video face replacement. *TOG (SIGGRAPH Asia)*, 2011. 2, 4, 8

[12] F. Diego, J. Serrat, and A. M. López. Joint spatio-temporal alignment of sequences. *Trans. on Multimedia*, 2013. 2

[13] T. Ezzat, G. Geiger, and T. Poggio. Trainable videorealistic speech animation. *TOG (SIGGRAPH)*, 2002. 2

[14] P. Garrido, L. Valgaerts, O. Rehmsen, T. Thormaehlen, P. Perez, and C. Theobalt. Automatic face reenactment. In *CVPR*, 2014. 2

[15] P. Garrido, L. Valgaerts, H. Sarmadi, I. Steiner, K. Varanasi, P. Perez, and C. Theobalt. VDub - modifying face video of actors for plausible visual alignment to a dubbed audio track. In *CGF (Eurographics)*, 2015. 2

[16] P. Garrido, L. Valgaerts, C. Wu, and C. Theobalt. Reconstructing detailed dynamic face geometry from monocular video. In *TOG (SIGGRAPH)*, 2013. 2

[17] A. Ghosh, G. Fyffe, B. Tunwattanapong, J. Busch, X. Yu, and P. Debevec. Multiview face capture using polarized spherical gradient illumination. *TOG (SIGGRAPH Asia)*, 2011. 2

[18] S. Grofit and Y. Lavner. Time-scale modification of audio signals using enhanced WSOLA with management of transients. *IEEE Transactions on Audio, Speech & Language Processing*, 2008. 7

[19] E. Hsu, K. Pulli, and J. Popovic. Style translation for human motion. *TOG (SIGGRAPH)*, 2005. 2

[20] I. Kemelmacher-Shlizerman, A. Sankar, E. Shechtman, and S. M. Seitz. Being John Malkovich. In *ECCV*, 2010. 2

[21] M. Klaudiny and A. Hilton. High-detail 3D capture and non-sequential alignment of facial performance. In *3DIMPVT*, 2012. 2

[22] C. Kuster, T. Popa, J.-C. Bazin, C. Gotsman, and M. Gross. Gaze correction for home video conferencing. *TOG (SIGGRAPH Asia)*, 2012. 2, 8

[23] H. Li, J. Yu, Y. Ye, and C. Bregler. Realtime facial animation with on-the-fly correctives. *TOG (SIGGRAPH)*, 2013. 2

[24] J. Liao, R. S. Lima, D. Nehab, H. Hoppe, and P. V. Sander. Semi-automated video morphing. *CGF (Eurographics Symposium on Rendering)*, 2014. 3, 5

[25] J. Liao, R. S. Lima, D. Nehab, H. Hoppe, P. V. Sander, and J. Yu. Automating image morphing using structural similarity on a halfway domain. *TOG*, 2014. 3

[26] W.-C. Ma, T. Hawkins, P. Peers, C.-F. Chabert, M. Weiss, and P. Debevec. Rapid acquisition of specular and diffuse normal maps from polarized spherical gradient illumination. In *Eurographics Symposium on Rendering*, 2007. 2

[27] W.-C. Ma, A. Jones, J.-Y. Chiang, T. Hawkins, S. Frederiksen, P. Peers, M. Vukovic, M. Ouhyoung, and P. Debevec. Facial performance synthesis using deformation-driven polynomial displacement maps. *TOG (SIGGRAPH Asia)*, 2008. 2

[28] P. Mermelstein. Distance measures for speech recognition, psychological and instrumental. *Pattern Recognition and Artificial Intelligence*, 1976. 4

[29] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. *TOG (SIGGRAPH)*, 2003. 3, 7

[30] T. Rhee, Y. Hwang, J. D. Kim, and C. Kim. Real-time facial animation from live video tracking. In *SCA*, 2011. 2

[31] P. Sand and S. J. Teller. Video matching. *TOG (SIGGRAPH)*, 2004. 2

[32] F. Shi, H. Wu, X. Tong, and J. Chai. Automatic acquisition of high-fidelity facial performances using monocular videos. *TOG (SIGGRAPH Asia)*, 2014. 2, 8

[33] M. Slaney, M. Covell, and B. Lassiter. Automatic audio morphing. In *ICASSP*, 1996. 7

[34] O. Sorkine. Least-squares rigid motion using SVD. Technical report, 2009. 4

[35] K. Sunkavalli, M. K. Johnson, W. Matusik, and H. Pfister. Multi-scale image harmonization. *TOG (SIGGRAPH)*, 2010. 3

[36] S. L. Taylor, M. Mahler, B. Theobald, and I. Matthews. Dynamic units of visual speech. In *SCA*, 2012. 2

[37] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *ICCV*, 1998. 3

[38] L. Valgaerts, C. Wu, A. Bruhn, H.-P. Seidel, and C. Theobalt. Lightweight binocular facial performance capture under uncontrolled lighting. *TOG (SIGGRAPH Asia)*, 2012. 2

[39] D. Vlasic, M. Brand, H. Pfister, and J. Popovic. Face transfer with multilinear models. *TOG (SIGGRAPH)*, 2005. 2

[40] O. Wang, C. Schroers, H. Zimmer, M. Gross, and A. Sorkine-Hornung. VideoSnapping: interactive synchronization of multiple videos. *TOG (SIGGRAPH)*, 2014. 2, 3, 4

[41] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *CVPR*, 2013. 3

[42] F. Xu, J. Chai, Y. Liu, and X. Tong. Controllable high-fidelity facial performance transfer. *TOG (SIGGRAPH)*, 2014. 2

[43] F. Yang, L. D. Bourdev, E. Shechtman, J. Wang, and D. N. Metaxas. Facial expression editing in video using a temporally-smooth factorization. In *CVPR*, 2012. 2, 8

[44] F. Zhou and F. De la Torre. Canonical time warping for alignment of human behavior. In *NIPS*, 2009. 2

[45] F. Zhou and F. De la Torre. Generalized time warping for multi-modal alignment of human motion. In *CVPR*, 2012. 2