

Eigen Appearance Maps of Dynamic Shapes

Adnane Boukhayma^(✉), Vagia Tsiminaki, Jean-Sébastien Franco,
and Edmond Boyer

LJK, Université Grenoble Alpes, Inria Grenoble Rhône-Alpes, Grenoble, France
{adnane.boukhayma,vagia.tsiminaki,jean-sebastien.franco,
edmond.boyer}@inria.fr

Abstract. We address the problem of building efficient appearance representations of shapes observed from multiple viewpoints and in several movements. Multi-view systems now allow the acquisition of spatio-temporal models of such moving objects. While efficient geometric representations for these models have been widely studied, appearance information, as provided by the observed images, is mainly considered on a per frame basis, and no global strategy yet addresses the case where several temporal sequences of a shape are available. We propose a per subject representation that builds on PCA to identify the underlying manifold structure of the appearance information relative to a shape. The resulting eigen representation encodes shape appearance variabilities due to viewpoint and motion, with Eigen textures, and due to local inaccuracies in the geometric model, with Eigen warps. In addition to providing compact representations, such decompositions also allow for appearance interpolation and appearance completion. We evaluate their performances over different characters and with respect to their ability to reproduce compelling appearances in a compact way.

1 Introduction

The last decade has seen the emergence of 3D dynamic shape models of moving objects, in particular humans, acquired from multiple videos. These spatio-temporal models comprise geometric and appearance information extracted from images, and they allow for subject motions to be recorded and reused. This is of interest for applications that require real 3D contents for analysis, free viewpoint and animation purposes and also for interactive experiences made possible with new virtual reality devices. This ability to now record datasets of subject motions bolsters the need for shape and appearance representations that make optimal use of the massive amount of image information usually produced. While dynamic shape representations have been extensively studied, from temporally coherent representations over a single sequence, to shape spaces that can encode

Electronic supplementary material The online version of this chapter (doi:[10.1007/978-3-319-46448-0_14](https://doi.org/10.1007/978-3-319-46448-0_14)) contains supplementary material, which is available to authorized users.

both pose and subject variabilities over multiple sequences and multiple subjects, appearance representations have received less attention in this context. In this paper, we investigate this issue.

Currently, appearance information is still most often estimated and stored once per frame, e.g. a texture map associated to a 3D model [1], and the leap to an efficient temporal appearance representation is still a largely open problem. This is despite the obvious redundancy with which the appearance of subjects is observed, across temporal frames, different viewpoints of the same scene, and often several sequences of the same subject performing different actions or motions. At the opposite of the spectrum, and given registered geometries, one can store only one texture for a sequence or even for a subject in several sequences, hence dramatically reducing sizes, but in so doing would drop the ability to represent desirable appearance variations, such as change in lighting or personal expression of the subject.

In this paper, we advance this aspect by providing a view-independent appearance representation and estimation algorithm, to encode the appearance variability of a dynamic subject, observed over one or several temporal sequences. Compactly representing image data from all frames and viewpoints of the subject can be seen as a non-linear dimensionality reduction problem in image space, where the main non-linearities are due to the underlying scene geometry. Our strategy is to remove these non-linearities with state-of-the-art geometric and image-space alignment techniques, so as to reduce the problem to a single texture space, where the remaining image variabilities can be straightforwardly identified with PCA and thus encoded as Eigen texture combinations. To this goal, we identify two geometric alignment steps (Fig. 1). First, we coarsely register geometric shape models of all time frames to a single shape template, for which we pre-computed a single reference surface-to-texture unwrapping. Second, to cope with remaining fine-scale misalignments due to registration errors, we estimate realignment warps in the texture domain. Because they encode low-magnitude, residual geometric variations, they are also advantageously decomposed using PCA, yielding Eigen warps. The full appearance information of all subject sequences can then be compactly stored as linear combinations of Eigen textures and Eigen warps. Our strategy can be seen as a generalization of the popular work of Nishino *et al.* [2], which introduces Eigen textures to encode appearance variations of a static object under varying viewing conditions, to the case of fully dynamic subjects with several viewpoints and motions.

The pipeline is shown to yield effective estimation performance. In addition, the learned texture and warp manifolds allow for efficient generalizations, such as texture interpolations to generate new unobserved content from blended input sequences, or completions to cope with missing observations due to *e.g.* occlusions. To summarize our main contribution is to propose and evaluate a new appearance model that specifically addresses dynamic scene modeling by accounting for both appearance changes and local geometric inaccuracies.

2 Related Work

Obtaining appearance of 3D models from images was first tackled from static images for inanimate objects, e.g. [2, 3], a case largely explored since e.g. [4, 5]. The task also gained interest for the case of subjects in motion, e.g. for human faces [6]. With the advent of full body capture and 3D interaction systems [1, 7] the task of recovering appearance has become a key issue, as the appearance vastly enhances the quality of restitution of acquired 3D models.

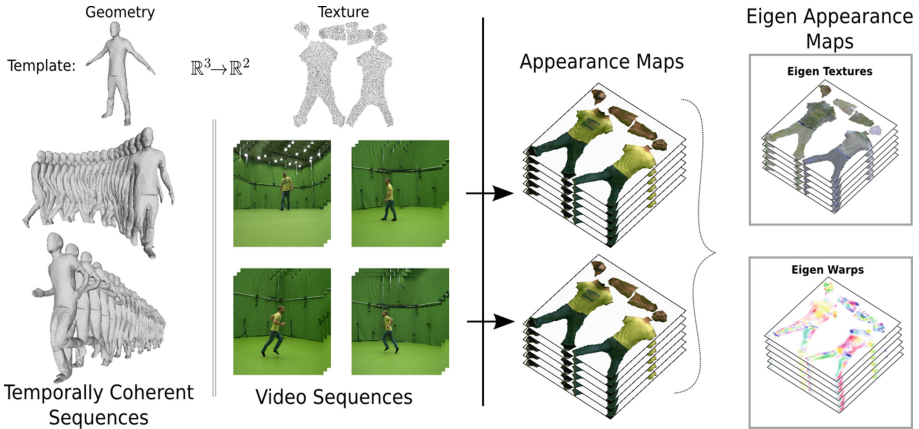


Fig. 1. Overview: Time consistent shape modeling provides datasets of appearance maps. Our proposed method exploits the manifold structure of these appearance information through PCA decomposition to generate the Eigen appearance maps relative to a shape.

A central aspect of the problem is how to represent appearance, while achieving a proper trade-off between storage size and quality. 3D capture traditionally generates full 3D reconstructions, albeit of inconsistent topology across time. In this context the natural solution is to build a representation per time frame which uses or maps to that instant's 3D model. Such per instant representations come in two main forms. View-dependent texturing stores and resamples from each initial video frame [8], eventually with additional alignments to avoid ghosting effects [9]. This strategy creates high quality restitutions managing visibility issues on the fly, but is memory costly as it requires storing all images from all viewpoints. On the other hand, one can compute a single appearance texture map from the input views in an offline process [1], reducing storage but potentially introducing sampling artifacts. These involve evaluating camera visibility and surface viewing angles to patch and blend the view contributions in a single common mapping space. To overcome the resolution and sampling limitations, 3D superresolution techniques have been devised that leverage the viewpoint multiplicity to build such maps with enhanced density and quality [10–12].

In recent years, a leap has been made in the representation of 3D surfaces captured, as they can now be estimated as a deformed surface of time-coherent topology [13, 14]. This in turns allows any surface unwrapping and mapping to be consistently propagated in time, however in practice existing methods have only started leveraging this aspect. Tsiminaki *et al.* [11] examines small temporal segments for single texture resolution enhancement. Volino *et al.* [15] uses a view-based multi-layer texture map representation to favour view-dependant dynamic appearance, using some adjacent neighbouring frames. Collet *et al.* [1] use tracked surfaces over small segments to improve compression rates of mesh and texture sequences. Methods are intrinsically limited in considering longer segments because significant temporal variability then appears due to light change and movement. While global geometry consistency has been studied [16–18], most such works were primarily aimed at animation synthesis using mesh data, and do not propose a global appearance model for sequences. In contrast, we propose an analysis and representation spanning full sequences and multiples sequences of a subject.

For this purpose, we build an Eigen texture and appearance representation that extends concepts initially explored for faces and static objects [2, 6, 19, 20]. Eigenfaces [19] were initially used to represent the face variability of a population for recognition purposes. The concept was broadened to built a 3D generative model of human faces both in the geometry and texture domains, using the fact that the appearance and geometry of faces are well suited to learning their variability as linear subspaces [6]. Cootes *et al.* [20] perform the linear PCA analysis of appearance and geometry landmarks jointly in their active appearance model. Nishino *et al.* [2] instead use such linear subspaces to encode the appearance variability of static objects under light and viewpoint changes at the polygon level. We use linear subspaces for full body appearance and over multiple sequences. Because the linear assumption doesn't hold for whole body pose variation, we use state of the art tracking techniques [21] to remove the non-linear pose component by aligning a single subject-specific template to all the subject's sequence. This in turn allows to model the appearance in a single mapping space associated to the subject template, where small geometric variations and appearances changes can then be linearly modeled.

3 Method

To eliminate the main geometric non-linearity, we first align sequence geometries to a single template shape and extract the texture maps of a subject over different motion sequences in a common texture space using a state-of-the-art method [11]. Other per-frame texture extractions may be considered. From these subject specific textures, Eigen textures and Eigen warps that span the appearance space are estimated. The main steps of the method below are depicted in Fig. 2 and detailed in the following sections.

1. Texture deformation fields that map input textures to, and from, their aligned versions are estimated using optical flows. Given the deformation fields, Poisson reconstruction is used to warp textures.

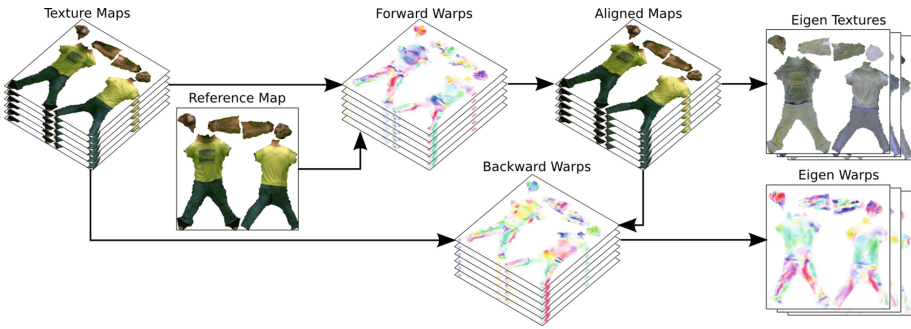


Fig. 2. Method pipeline from input textures (left) to eigen maps (right).

2. PCA is applied to the aligned maps and to the texture warps to generate the Eigen textures and the Eigen warps that encode the appearance variations due to, respectively, viewpoint, illumination, and geometric inaccuracies in the reference model.

Hence, The main modes of variation of aligned textures and deformation fields, namely Eigen textures and Eigen warps respectively, span the appearance space in our representation. The main steps of this method are depicted in Fig. 2 and detailed in the following sections.

Note that due to texture space discretization, the warps between textures are not one-to-one and, in practice, two separate sets of warps are estimated. Forward warps map the original texture maps to the reference map. Backward warps map the aligned texture maps back to the corresponding input textures (see Fig. 2).

3.1 Aligning Texture Maps

Appearance variations that are due to viewpoint and illumination changes are captured through PCA under linearity assumption for these variations. To this purpose, textures are first aligned in order to reduce geometric errors resulting from calibration, reconstruction and tracking imprecisions. Such alignment is performed using optical flow, as described below, and with respect to a reference map taken from the input textures. An exhaustive search of the best reference map with the least total alignment error over all input textures is prohibitive since it requires N^2 alignments given N input textures. We follow instead a medoid shift strategy over the alignment errors.

The alignment algorithm (see Algorithm 1) first initializes the reference map as one texture from the input set. All texture maps are then aligned to this reference map, and the alignment error is computed as the cumulative sum of squared pixel differences between the reference and the aligned texture maps. The medoid over the aligned texture maps, with respect to alignment error, then identifies the new reference map. These two steps, alignment and medoid shift, are iterated until the total alignment error stops decreasing.

Data: Texture maps $\{I_k\}_{k \in [1..N]}$

Result: Reference map A_{ref} , aligned textures A_k

A_{ref}, e_0 initializations;

while $e_i < e_{i-1}$ **do**

 Compute alignment warps: $\{w_k\}_{k \in [1..N]}$ s.t. $A_{ref} \approx I_k(x + w_k)$;

 Align texture maps: $A_k = I_k(x + w_k)$;

 Update alignment error: $e_i = \sum_k \|A_k - A_{ref}\|^2$;

 Set A_{ref} as the texture that gives the medoid of the aligned textures:

$A_{ref} = I_{k_0}$ s.t. $k_0 = \arg \min_k \sum_l \|A_k - A_l\|^2$;

end

Algorithm 1. Texture alignment with iterative reference map selection.

Dense Texture Correspondence with Optical Flow. The warps $\{w_k\}$ in the alignment algorithm, both forward and backward in practice, are estimated as dense pixel correspondences with an optical flow method [22]. We mention here that the optical flow assumptions: brightness consistency, spatial coherency and temporal persistence, are not necessarily verified by the input textures. In particular, the brightness consistency does not hold if we assume appearance variations with respect to viewpoint and illumination changes. To cope with this in the flow estimation, we use histogram equalization as a preprocessing step, which presents the benefit of enhancing contrast and edges within images. Additionally, local changes in intensities are reduced using bilateral filtering, which smooths low spatial-frequency details while preserving edges.

Texture Warping. Optical flows give dense correspondences $\{w\}$ between the reference map and the input textures. To estimate the aligned textures $\{A\}$, we cast the problem as an optimization that seeks the texture map which, once moved according to w , best aligns with the considered input texture both in the color and gradient domains. Our experiments show that solving over both color and gradient domains significantly improves results as it tends to better preserve edges than with colors only. This is also demonstrated in works that use the Poisson editing for image composition, e.g. [23, 24] or interpolation, e.g. [25, 26]. We follow here a similar strategy.

We are given an input texture map I , a dense flow w from A_{ref} to I , and the gradient image ∇I . The aligned texture A of I with respect to A_{ref} is then the map that minimizes the following term:

$$E(A) = \Sigma_x \|\nabla^2 A(x) - \vec{\nabla} \cdot \nabla I(x + w)\|^2 + \lambda \|A(x) - I(x + w)\|^2, \quad (1)$$

where ∇^2 is the Laplacian operator, $\vec{\nabla} \cdot$ the divergence operator, and x denotes pixel locations in texture maps. The weight λ balances the influence of color and gradient information. In our experiments, we found that the value 0.02 gives the best results with our datasets.



Fig. 3. Poisson versus direct texture warping.

Using a vector image representation, the above energy can be minimized by solving, in the least-squares sense, the overdetermined $2N \times N$ system below, where N is the active region size of texture maps:

$$\begin{pmatrix} L \\ \Lambda \end{pmatrix} A = \begin{pmatrix} \vec{\nabla} \cdot \nabla I(x+w) \\ \Lambda I(x+w) \end{pmatrix}, \quad (2)$$

where L is the linear Laplacian operator and $\Lambda = \text{diag}_N(\lambda)$. A solution for A is easily found by solving the associated normal equations:

$$(L^T L + \Lambda^2) A = L^T \vec{\nabla} \cdot \nabla I(x+w) + \Lambda^2 I(x+w). \quad (3)$$

Figure 3 shows an example where a texture map is warped, given a warp field, using both direct pixel remapping and Poisson warping. The latter strategy achieves visually more compelling and edge preserving results.

3.2 Eigen Textures and Eigen Warps

Once the aligned textures and the warps are estimated, we can proceed with the statistical analysis of appearances. Given the true geometry of shapes and their motions, texture map pixels could be considered as shape appearance samples over time and PCA applied directly to the textures would then capture appearance variability. In practice, incorrect geometry causes distortions in the texture space and textures must be first aligned before any statistical analysis. In turn, de-alignment must be also estimated to map the aligned textures back to their associated input textures (see Fig. 2). And these backward warps must be part of the appearance model to enable appearance reconstruction. In the following, warps denote the backward warps. Also, we consider vector representations of the aligned texture maps and of the warps. These representations include only pixels that fall inside active regions within texture maps. We perform Principal Component Analysis on the textures and on the warp data separately to find the orthonormal bases that encode the main modes of variation in the texture space and in the warp space independently. We refer to vectors spanning the texture space as Eigen textures, and to vectors spanning the warp space as Eigen warps.

Let us consider first texture maps. Assume N is the dimension of the vectorized representation of active texture elements, and F the total number of frames available for the subject under consideration. To give orders of magnitude for our datasets, $N = 22438995$ and $F = 207$ for the TOMAS dataset, and $N = 25966476$

and $F = 290$ for the CATY dataset that will be presented in the next section. We start by computing the mean image \bar{A} and the centered data matrix M from aligned texture maps $\{A_i\}_{i \in [1..F]}$:

$$\bar{A} = \frac{1}{F} \sum_k A_k, \quad M = \begin{bmatrix} | & & | \\ A_1 - \bar{A} & \dots & A_F - \bar{A} \\ | & & | \end{bmatrix}. \quad (4)$$

Traditionally, the PCA basis for this data is formed by the Eigen vectors of the covariance matrix MM^T , of size $N \times N$, but finding such vectors can easily become prohibitive as a consequence of the texture dimensions. However, it appears that the non zero eigen values of MM^T are equal to the non zero Eigen values of $M^T M$, of size $(F \times F)$ this time, and that they are at most: $\min(F, N) - 1$. Based on this observation, and since $F \ll N$ in our experiments, we solve the characteristic equation $\det(MM^T - \alpha I_N) = 0$ by performing Singular Value Decomposition on the matrix $M^T M$, as explained in [19]:

$$M^T M = D \Sigma D^T, \quad D = \begin{bmatrix} | & & | \\ V_1 & \dots & V_F \\ | & & | \end{bmatrix} \quad (5)$$

where D contains the $(F - 1)$ orthonormal Eigen vectors $\{V_i\}$ of $M^T M$, and $\Sigma = \text{diag}(\alpha_i)_{1 \leq i \leq F}$ contains the eigen values $\{\alpha_i\}_{1 \leq i \leq F-1}$. We can then write:

$$M^T M V_i = \alpha_i V_i, \quad i \in [1..F - 1], \quad (6)$$

and hence:

$$MM^T \underbrace{M V_i}_{T_i} = \alpha_i \underbrace{M V_i}_{T_i}, \quad i \in [1..F - 1], \quad (7)$$

where T_i are the Eigen vectors of MM^T and therefore form the orthonormal basis of the aligned texture space after normalization, namely the Eigen textures.

In a similar way, we obtain the mean warp \bar{w} and the orthonormal basis of the warp space $\{W_i\}_{1 \leq i \leq F-1}$, the Eigen warps.

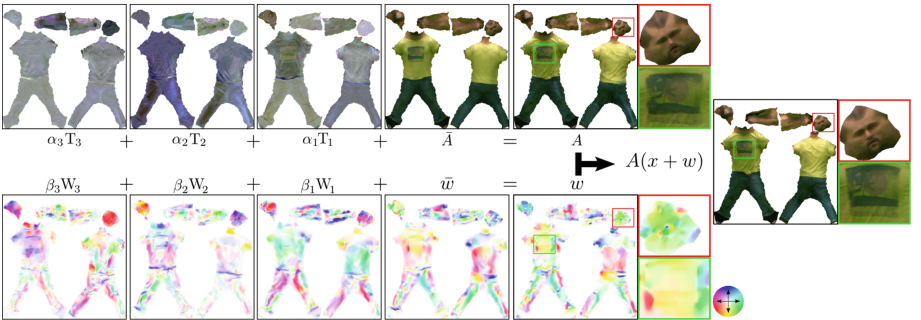


Fig. 4. Texture map generation by linear combination.

3.3 Texture Generation

Given the Eigen textures and the Eigen warps, and as shown in Fig. 4, a texture can be generated by first creating an aligned texture by linearly combining Eigen textures and second de-aligning this new texture using another linear combination of the Eigen warps.

4 Performance Evaluation

To validate the estimation quality of our method, we apply our estimation pipeline to several datasets, project and warp input data using the built eigen-spaces, then evaluate the reconstruction error. To distinguish the different error sources, we evaluate this error both in texture space before projection, and in image domain by projecting into the input views, as compared to the original views of the object and the texture before any reconstruction in texture space, estimated in our pipeline using [11]. For the image error measurement, we use the 3D model that was fitted to the sequence, as tracked to fit the test frames selected [21], and render the model as textured with our reconstructed appearance map, using a standard graphics pipeline. In both cases, we use the structural similarity index (SSIM) [27] as metric to compare to the original. All of our SSIM estimates are computed in the active regions of the texture and image domains, that is on the set of texels actually mapped to the 3D model in the texture domain, and only among actual silhouette pixels in the image domain.

We study in particular the compactness and generalization abilities of our method, by examining the error response as a function of the number of eigen components kept after constructing the linear subspaces, and the number of training images selected. For all these evaluations, we also provide the results for a naive PCA strategy, where only a set of eigen appearance maps are built in texture space and use to project and reconstruct textures, to show the performance contribution of including the Eigen warps.

For validation, we used two multi-sequence datasets: (1) the TOMAS dataset which consists of 4 different sequences left, right, run and walk with 207 total number of frames and 68 input views each captured at resolution 2048×2048 pixels per frame; and (2) the CATY dataset: low, close, high and far jumping sequences with 290 total number of frames and 68 input views each captured at resolution 2048×2048 pixels per frame.

4.1 Estimation Quality and Compactness

We study the quality and compactness of the estimated representation by plotting the SSIM errors of reconstructed texture and image estimates of our method against naive PCA, for the two multi-sequence datasets (Fig. 5). Note that all texture domain variability could be trivially represented by retaining as many Eigen textures as there are input images, thus we particularly examine how the quality degrades with the fraction of Eigen components kept. In the case of image

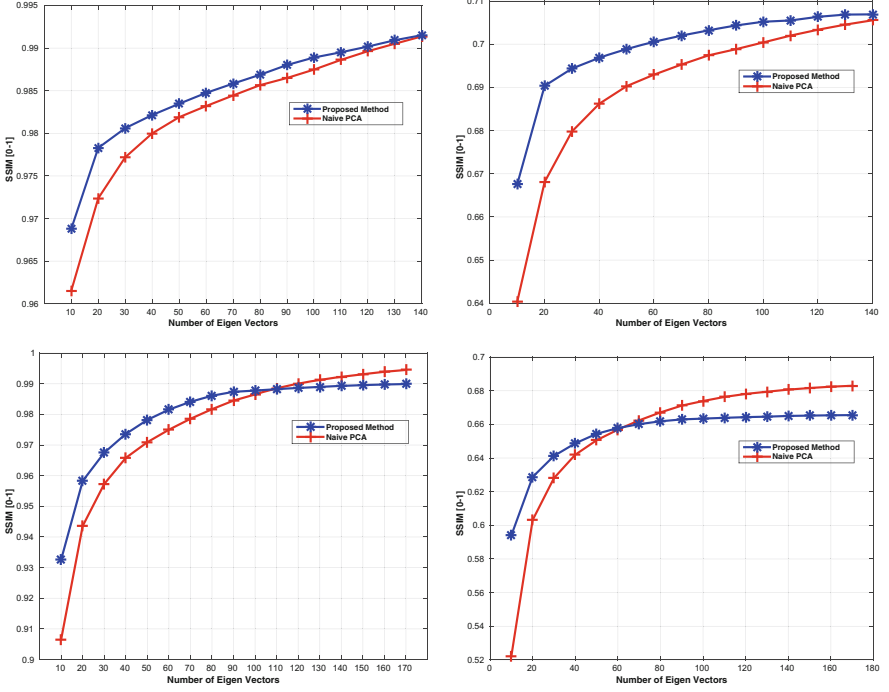


Fig. 5. Reconstruction Error for TOMAS and CATY Dataset from top to down in Texture and Image Domain from left to right.

domain evaluations, we plot the average SSIM among all viewpoints. Our method outperforms naive PCA in image and texture domains on both datasets, achieving higher quality with a lower number of Eigen components, and only marginally lower quality as the number of components grows, where the method would be anyway less useful. Higher number of Eigen components marginally favors naive PCA, because naive PCA converges to input textures when increasing the Eigen textures retained by construction, whereas our method hits a quality plateau due to small errors introduced by texture warp estimation and decomposition. For both datasets, virtually no error (0.98 SSIM) is introduced by our method in the texture domain with as low as 50 components, a substantially low fraction compared to the number of input frames (207 and 290). This illustrates the validity of the linear variability hypothesis in texture domain. The error is quite higher in the image domain (bounded by 0.7) for both our method and naive PCA, because measurements are then subject to fixed upstream errors due to geometric alignments, projections and image discretizations. Nevertheless, visually indistinguishable results are achieved with 50 Eigen components (images and warps), with a significant compactness gain.

4.2 Generalization Ability

In the previous paragraph, we examined the performance of the method by constructing an Eigen space with all input frames. We here evaluate the ability of the model to generalize, *i.e.* how well the method reconstructs textures from input frames under a reduced number of examples that don't span the whole input set. For this purpose, we perform an experiment using a varying size training set, and a test set from frames not in the training set. We use a training set comprised of randomly selected frames spanning 0 % to 60 % of the total number of frames, among all sequences and frames of all datasets, and plot the error of projecting the complement frames on the corresponding Eigen space (Fig. 6). The experiment shows that our representation produces a better generalization than naive PCA, *i.e.* less training frames need to be used to reconstruct a texture and reprojections of equivalent quality. For the TOMAS dataset, one can observe that less than half training images are needed to achieve similar performance in texture space, and a quarter less with the CATY dataset.

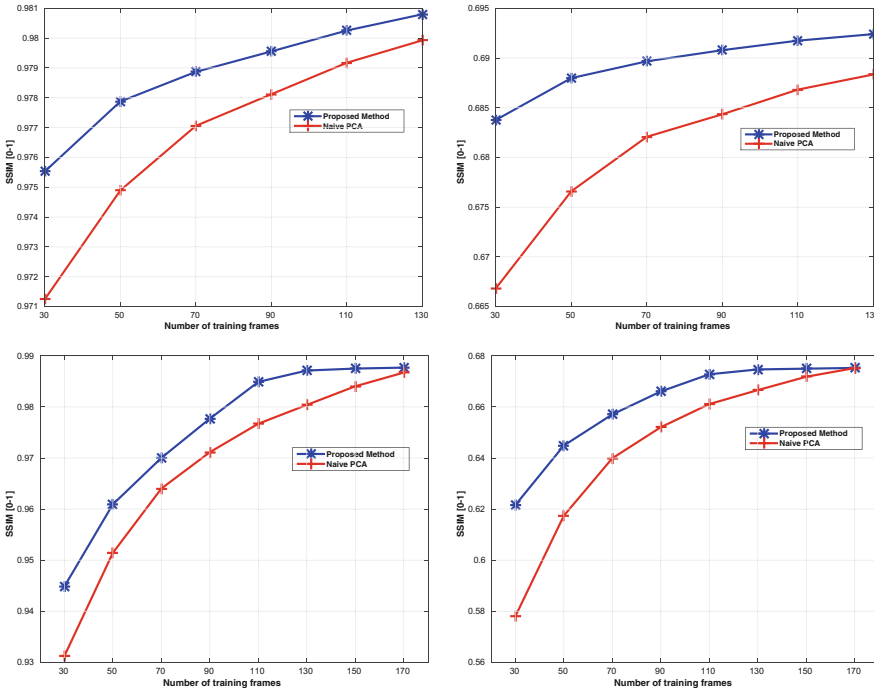


Fig. 6. Generalization Error for THOMAS and CATY Dataset from top to down in Texture and Image Domain from left to right.

5 Applications

We investigate below two applications of the appearance representation we propose. First, the interpolation between frames at different time instants and second, the completion of appearance maps at frames where some appearance information is lacking due to occlusions or missing observations during the acquisition. Results are shown in the following section and the supplementary video.

5.1 Interpolation

In our framework, appearance interpolation benefits from the pre-computed warps and the low dimensionality of our representation to efficiently synthesize compelling new appearances with reduced ghosting-artefacts. It also easily enables extension of appearance interpolation from pairwise to multiple frames. Assume that shapes between two given frames are interpolated using a standard non-linear shape interpolation, for instance [28]. Consider then the associated aligned textures and associated warps at the given frames. We perform a linear interpolation in the Eigen texture and Eigen warp spaces respectively by blending the projection coefficients of the input appearance maps. Poisson warping, as introduced in Sect. 3.1 is used to build de-aligned interpolated texture with the interpolated backward warp. Figure 7 compares interpolation using our pipeline to a standard linear interpolation for 4 examples with the CATY and TOMAS datasets. Note that our method is also linear but benefits from the alignment performed in the texture space to reduce interpolation artefacts, as well as from the simplified computational aspects since interpolation applies to projection coefficients only.

5.2 Completion

As mentioned earlier, appearance maps can be incomplete due to acquisition issues. For instance, as shown in Fig. 8, during the running sequence the actor TOMAS bends his knees in such a way that the upper parts of his left and right shins become momentarily hidden to the acquisition system. This results in missing information for those body parts in the texture maps and over a few frames. Such an issue can be solved with our texture representation by omitting the incomplete frames when building our appearance representations, and then projecting these incomplete appearance maps in the Eigen spaces and reconstructing them using the projection coefficients and Poisson texture warping. Figure 8 shows two examples of this principle with occluded regions. Note however, that while effectively filling gaps in the appearance map, this completion might yet lose appearance details in regions of the incomplete map where information is not duplicated in the training set.



Fig. 7. Interpolation examples using linear interpolation (left) and our pipeline (right). From left to right: Input frames, Interpolated models, and a close-up on the texture maps (top) and the rendered images (bottom).



Fig. 8. Completion examples. From left to right: Input and completed models, close-up on input and completed texture maps (top) and rendered images (bottom).

6 Conclusion

We have presented a novel framework to efficiently represent the appearance of a subject observed from multiple viewpoints and in different motions. We propose a straightforward representation which builds on PCA and decomposes into Eigen textures and Eigen warps that encode, respectively, the appearance variations due to viewpoint and illumination changes and due to geometric modeling imprecisions. The framework was evaluated on 2 datasets and with respect to: (i) its ability to accurately reproduce appearances with compact representations; (ii) its ability to resolve appearance interpolation and completion tasks. In both cases, the interest of a global appearance model for a given subject was demonstrated. Among the limitations, the representation performances are dependent on the underlying geometries. Future strategies that combine both shape and appearance information would thus be of particular interest. The proposed model could also be extended to global representations over populations of subjects.

References

1. Collet, A., Chuang, M., Sweeney, P., Gillett, D., Evseev, D., Calabrese, D., Hoppe, H., Kirk, A., Sullivan, S.: High-quality streamable free-viewpoint video. *ACM Trans. Graph.* **34**, 69:1–69:13 (2015)
2. Nishino, K., Sato, Y., Ikeuchi, K.: Eigen-texture method: appearance compression and synthesis based on a 3D model. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**, 1257–1265 (2001)
3. Debevec, P.E., Taylor, C.J., Malik, J.: Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach. In: *ACM SIGGRAPH 1996* (1996)
4. Lempitsky, V.S., Ivanov, D.V.: Seamless mosaicing of image-based texture maps. In: *CVPR* (2007)
5. Waechter, M., Moehrle, N., Goesele, M.: Let there be color! Large-scale texturing of 3D reconstructions. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8693, pp. 836–850. Springer, Heidelberg (2014). doi:[10.1007/978-3-319-10602-1_54](https://doi.org/10.1007/978-3-319-10602-1_54)
6. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3D faces. In: *ACM SIGGRAPH 1996* (1999)
7. Carranza, J., Theobalt, C., Magnor, M.A., Seidel, H.P.: Free-viewpoint video of human actors. *ACM Trans. Graph.* **22**, 569–577 (2003)
8. Zitnick, C., Kang, S., Uyttendaele, M., Winder, S., Szeliski, R.: High-quality video view interpolation using a layered representation. In: *ACM SIGGRAPH 2004* (2004)
9. Eisemann, M., De Decker, B., Magnor, M., Bekaert, P., de Aguiar, E., Ahmed, N., Theobalt, C., Sellent, A.: Floating textures. *Comput. Graph Forum (Proc. of Eurographics)* **27**, 409–418 (2008)
10. Tung, T.: Simultaneous super-resolution and 3D video using graph-cuts (2008)
11. Tsiminaki, V., Franco, J.S., Boyer, E.: High resolution 3D shape texture from multiple videos. In: *CVPR* (2014)
12. Goldlücke, B., Aubry, M., Kolev, K., Cremers, D.: A super-resolution framework for high-accuracy multiview reconstruction. *Int. J. Comput. Vis.* **106**, 172–191 (2014)
13. de Aguiar, E., Stoll, C., Theobalt, C., Ahmed, N., Seidel, H.P., Thrun, S.: Performance capture from sparse multi-view video. *ACM Trans. Graph.* **27**, 98:1–98:10 (2008)
14. Cagniart, C., Boyer, E., Ilic, S.: Free-from mesh tracking: a patch-based approach. In: *CVPR* (2010)
15. Volino, M., Casas, D., Collomosse, J., Hilton, A.: Optimal representation of multiple view video. In: *BMVC* (2014)
16. Boukhayma, A., Boyer, E.: Video based animation synthesis with the essential graph. In: *3DV* (2015)
17. Casas, D., Tejera, M., Guillemaut, J.Y., Hilton, A.: 4D parametric motion graphs for interactive animation. In: *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (2012)
18. Casas, D., Volino, M., Collomosse, J., Hilton, A.: 4D video textures for interactive character appearance. *Comput. Graph. Forum (Proc. of Eurographics)* **33**, 371–380 (2014)
19. Turk, M., Pentland, A.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**, 71–86 (1991)

20. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(6), 681–685 (2001)
21. Allain, B., Franco, J.S., Boyer, E.: An efficient volumetric framework for shape tracking. In: *CVPR* (2015)
22. Sanchez Prez, J., Meinhardt-Llopis, E., Facciolo, G.: TV-L1 optical flow estimation. *Image Process. On Line* **3**, 137–150 (2013)
23. Pérez, P., Gangnet, M., Blake, A.: Poisson image editing. *ACM Trans. Graph.* **22**(3), 313–318 (2003)
24. Chen, T., Zhu, J.Y., Shamir, A., Hu, S.M.: Motion-aware gradient domain video composition. *IEEE Trans. Image Process.* **22**(7), 2532–2544 (2013)
25. Linz, C., Lipski, C., Magnor, M.: Multi-image interpolation based on graph-cuts and symmetric optical flow (2010)
26. Mahajan, D., Huang, F.C., Matusik, W., Ramamoorthi, R., Belhumeur, P.N.: Moving gradients: a path-based method for plausible image interpolation. *ACM Trans. Graph.* **28**(3), 1–11 (2009)
27. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.* **13**, 600–612 (2004)
28. Xu, D., Zhang, H., Wang, Q., Bao, H.: Poisson shape interpolation. In: *ACM Symposium on Solid and Physical Modeling* (2005)