Pose and expression-coherent face recovery in the wild

Xavier P. Burgos-Artizzu Joaquin Zepeda François Le Clerc Patrick Pérez
Technicolor, Cesson-Sévigné, France
xavier.burgos,joaquin.zepeda,francois.leclerc,patrick.perez@technicolor.com

Abstract

We present a novel method to recover images of faces, particularly when large spatial regions of the face are unavailable due to data losses or occlusions. In contrast with previous work, we do not make assumptions on the data neither during training nor testing (such as assuming that the person was seen before or that all faces are perfectly aligned and have identical head pose, expression, etc.). Instead, we propose to tackle the problem in a purely unsupervised way, leveraging a large face dataset. During training, first we cluster faces based on their landmark's positions (obtained by an automatic face landmark estimator). Then, we model the face appearance for each group using sparse coding with learned dictionaries, with one dictionary per cluster. At test time, given a face to recover, we find its belonging cluster and occluded area and restore missing pixels by applying the group-specific sparse appearance representation learned during training. We show results on two "in the wild" datasets. Our method shows promising results on challenging faces and our sparse coding approach outperforms prior subspace learning techniques.

1. Introduction

Human faces captured in real conditions often are partially hidden by occlusions due to a subject's interaction with the environment or factors such as wearing sunglasses, hats, long hair, etc. Furthermore, data transfer or storage errors can cause large areas of the image to be unavailable. Our proposed approach is able to recover the face occluded/unavailable regions in a seaming-less manner, resulting in images where the face is fully visible, Fig. 1.

The proposed approach has a wide range of applications. Examples include image forensics (recovering lost data), video-conferencing (recovering a person's smile even if the mouth is hidden) or image editing, to name just a few. This method can also be useful as a pre-processing to facial recognition tasks which are hindered by heavy occlusions (face alignment [5], expression detection [28], identity recognition [15], face retrieval [32], etc.).

Prior work [12, 21, 31, 41, 13, 25, 7, 19, 24, 26] focuses on performing face recovery in a pose and expression-neutral scenario, very often using several pictures of the same person for training and assuming all faces are perfectly aligned. In real-world conditions, however, faces can show a wide variety of head poses and expressions, and one cannot assume that the subject was seen before nor that faces are all perfectly aligned and scaled.

Figure 1: Example results. Our method is able to reconstruct a face presenting missing data or occlusions. In contrast to prior work, it can do so for a variety of head poses, identities and expressions, and does not require the person to have been seen before. Moreover, the reconstruction is able to preserve closely the person's original expression exploiting spatial correlations between different parts of the face during human display of emotion.

In this paper we propose a method able to recover faces regardless of its head poses, expressions or identities while avoiding any prior assumption on the data and dealing with the faces directly in an unsupervised fashion. We also specifically design our approach to preserve the original subject's expression. This is important since it is well known that facial spatial dependencies are highly correlated to face expressions [14].

In order to achieve our goal, we propose to leverage a large existing "in the wild" face database containing $> 70K$

images of 530 celebrities, called FaceScrub [32]. The proposed approach is as follows: 1) cluster the training faces based on their normalized landmark's positions (result of an automatic face landmarking method [5]) and 2) model facial appearance inside each cluster using sparse coding.

The landmark-based clustering ensures that faces are grouped according to their similarity in terms of head pose and the overall shape of facial parts (expressions). Then, the cluster-specific modeling of appearance will exploit subtle spatial dependencies to achieve in-painting of missing face pixels in a seaming-less manner.

The contributions of this paper are several:

1. A novel method to recover lost pixels from a face image. Unlike prior work, our method does not need identity, pose nor expression to be known a priori neither during training nor testing. Moreover, as far as we know, this is the first method that performs an expression-based recovery, leveraging the well known fact that some natural human expressions are manifested in all parts of the face (*e.g.* both the eyes and the mouth take a particular form when one smiles).

2. A hybrid clustering / sparse coding method wherein a dictionary is learned from the images corresponding to each landmark cluster. The fact that images inside each cluster are very similar means that the implicit dimensionality of the underlying data is low, and we successfully exploit this by using dictionaries that are highly undercomplete (e.g., 80 atoms for a signal space with several thousand dimensions), making it further possible to use signal vectors with supports covering the whole face region.

3. Coupling of our approach with modern face landmarking approaches [5, 17] able to detect occlusions, so that the region to be recovered can be estimated on-line.

4. Close to real-time performance. Our method runs at speeds close to $10 fps$ on a standard PC using unoptimized Python code.

2. Prior work

Faces have been since the beginning a popular object on which to benchmark novel subspace learning techniques, aimed at improving representation power and robustness compared to classical techniques such as PCA [33]. Some examples are Robust-PCA [12], Aproximated Principal Gradient (APG) [25], Singular Value Thresholding (SVT) [7] or Euler-PCA [26]. In these works, the training and test subjects are usually the same.

Hwang et al. [21] were among the first to tackle the problem of recovering partially occluded faces in a realistic scenario where training and test faces are different and a morphable model is used to remove face shape variations. They proposed to prototype faces as a PCA-based projection of both shapes and textures, much like in the original AAM formulation [9]. Similarly, [31, 41, 19] proposed to combine morphable models with modern dimensionality reduction techniques such as the ones mentioned above. Finally, [13] proposed a Bayesian framework that also allows the automatic detection of face occlusions.

The most crucial difference between these approaches and ours is the assumptions made on the data. They generally use controlled-scenario datasets such as FERET [34] and AR [29] and assume that head pose is constant, that faces were either previously aligned or that ground-truth facial landmarks are known, that all faces have a neutral expression and often even train on each test subject.

Instead, we use images taken "in the wild" and make no prior assumptions whatsoever. We use a face landmarking method to estimate landmarks automatically and rely on an unsupervised clustering to group similar faces together before learning each group's appearance. Furthermore, we separate subjects into two clearly separated train/test sets, never training on the subject whose face needs to be recovered at test time. We further contribute by adapting sparse coding techniques to the task, see Section 3.2. Finally, we propose to couple our approach with modern face landmarking methods able to estimate occlusion to build on-line the face recovery mask.

For completeness, we also discuss some of the prior work on other different areas of computer vision where one can find some common ground with the techniques discussed in this paper. However, since they do not fully overlap with this work, we address them briefly due to space constraints.

Inpainting This work is somewhat related to prior work on image editing by example-based inpainting [11], where an image region is replaced in a seaming-less manner by stitching together fragments from the same scene. These methods, however, struggle with very large regions, especially in presence of highly semantic content, which is typically the situation met for face recovery. In [18], these limitations of example-based inpainting are circumvented by relying on an external database: large scene occlusions are completed using images from similar scenes. While this approach bears some connection to our work, it lacks appearance modeling that is required to operate on very structured semantic visual content like faces.

Face expression transfer Another example is facial expression transfer [38, 20]. However, these approaches deal with fully unoccluded faces and the goal is usually that of transferring expressions across individuals, or from a video stream into a 3D animated model by estimating 3D facial landmarks. Instead, our approach allows us to recover the

original expression in large occluded regions of the face, based on what remains visible in the face.

Sparse coding applied to faces Sparse coding has been used to address a variety of face-related tasks. The work of [40] addresses the related face super-resolution problem (known as *face hallucination*) using a hybrid whole-face NMF factorization (without geometrical normalization) followed by block-by-block sparse coding using a dictionary consisting of examples. The work of [39] uses a whole-face dictionary of examples but to address the face recognition problem. In [22], the authors address the same application but instead use a learned dictionary constrained to have spatially-localized atoms. The work of [4] uses a piece-wise affine physiognomy normalization for the task of expression-neutral face compression, yet the authors again use 8×8 blocks as signal vectors with dictionaries learned for each spatial position.

The vast majority of algorithms employing sparse coding on images operate on small image blocks (*e.g.* 8×8). Besides the complexity issues related to larger dimensionality [36, 43], the reason for this is that spatial redundancy decreases when using larger block sizes, making it difficult to use dictionaries of practical sizes. Yet images of faces, particularly when the faces are geometrically normalized using piece-wise affine warping, enjoy high spatial dependency, making it possible to operate on the entire image. Indeed [39] exploits this advantage to carry out sparse-coding-based face recognition, and [4] employs a related redundancy-enhancing approach for face compression.

Face landmark estimation The first step to our method is based on face landmark estimation. Early work on the topic includes Active Contours Models [23], Template Matching [42], Active Shape Models (ASM) [10] and Active Appearance Models (AAM) [9]. Popular modern approaches involve first detecting the object parts independently and then estimating shape through flexible parts models [16, 17]. Another family of approaches is that which tackles shape estimation as a regression problem, learning regressors that directly predict the object shape or the location of its parts, starting from a raw estimate of its position [8, 5, 35]. These methods are fast and precise, being able to deal with large amounts of occlusion. We use the approach in [5] due to its speed, capability of detecting occlusion and availability of code on-line.

3. Proposed approach

Fig. 2 shows the outline of the proposed approach. Our method relies heavily on the automatic estimation of face landmarks, for which we use Robust Cascaded Pose Regression (RCPR) [5] re-trained using the exhaustive *300 faces-in-the-wild challenge* dataset [37].

Once face landmarks are detected, we normalize all faces to be the same size and cluster faces based on their landmark distance, see Sec. 3.1. Then, we learn a separate appearance model for each cluster using sparse coding, see Sec. 3.2. At test time, given a previously unseen occluded image, we detect the occlusion mask in real-time from the output of RCPR and recover the occluded pixels by applying the learned appearance model, see Sec. 3.3.

3.1. Face clustering

For face recovery to work, the key is to be able to reconstruct the missing pixels using an appearance model (*a.k.a.* texture model) learned on faces that closely resemble the test face. Prior work supposes that all images are aligned (or ground-truth landmarks are available), that faces show neutral expressions and that the face to be recovered has been seen during training. Under those conditions, one can apply subspace learning techniques to learn a person-specific texture model, maximizing the chances that it will be able to recover the missing pixels when occluded.

When all the above suppositions about the data are removed, however, the task becomes daunting. Learning a rich and at the same time generalizable global texture model from any set of faces is practically unfeasible. This is a well known fact, which is precisely the reason behind AAM's poor generalization performance compared to methods using more localized texture models such as ASM's [30].

To avoid this issue, we propose to previously cluster faces based on their similarity and then learn a separate appearance model for each group (explained in Section 3.2).

Interestingly, face landmarks are a natural way of encoding both the head pose and expressions [8, 6]. Therefore, performing an unsupervised clustering using the distance between (position and scale normalized) landmarks can lead to groups with similar overall appearance.

Given P 2-D landmark locations (typically around chin, eyes, mouth, and nose) that implicitly encode the shape of the face as $\mathcal{S} = [\mathbf{a}, \mathbf{b}]$ where $\mathbf{a}, \mathbf{b} \in \mathbb{R}^P$ we derive a scale-normalized version $\hat{\mathcal{S}} = [\hat{\mathbf{a}}, \hat{\mathbf{b}}]$, with

$$\hat{\mathbf{a}} = \frac{\mathbf{a} - \min(\mathbf{a})}{\max(\mathbf{a})}, \quad \hat{\mathbf{b}} = \frac{\mathbf{b} - \min(\mathbf{b})}{\max(\mathbf{b})}. \tag{1}$$

Here $min(.)$ and $max(.)$ are the minimum and maximum values of a vector and we abuse notation of the subtract and division operations to represent element-wise operation.

Once landmarks are normalized, we apply k-means algorithm to all N training landmarks $\hat{\mathcal{S}}_i, i \in \{1 \cdots N\}$. Instead of setting the parameter K (number of clusters), we prefer to fix it to a large number (*e.g.* 10^3) and enforce a minimum cluster size SZ. Note that while both parameters have the same role (higher SZ will cause fewer clusters to be found, identical to setting a lower K), SZ is much more intuitive

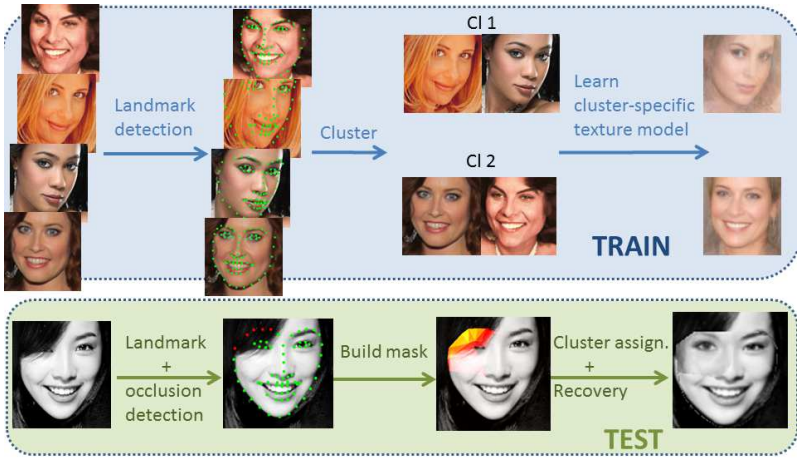


Figure 2: Method outline. During training our method clusters all faces based on their normalized landmark's positions and models facial appearance inside each cluster using sparse coding. During testing the belonging cluster is found and the group-specific learned model is applied to recover occluded pixels, estimated on-line.

Figure 3: Randomly-selected faces corresponding to some example clusters (row-wise), using minimum cluster size $SZ = 10$.

since it will control the amount of training images we will be using to build our appearance model.

Fig. 3 shows an example of some of the clusters found by kmeans using $SZ = 10$ [1]. As it can be seen, faces inside a cluster are correctly aligned and show similar head pose and expression, exactly as needed for the next step.

3.2. Sparse encoding of face appearance

Once training faces are clustered, we learn for each group an appearance model that we will later apply to recover missing pixels from testing faces. Cluster-specific model relies on first normalizing training faces in the cluster, both geometrically (using an invertible piecewise-affine warp toward average shape) and photometrically (via pixel-wise centering of intensities). Resulting aligned face textures are then sparsely encoded based on a learned dictionary. We describe these steps in the present section.

[1] We manually separate men and women during training and testing

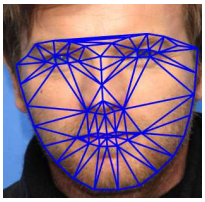


Figure 4: Visualization of the delaunay triangulation from face landmarks

Cluster-dependent face alignment: The first step of our modeling is to apply a geometrical normalization, warping all images belonging to the same cluster to show an aligned face shape. We first define a *standard face shape* $\bar{\mathcal{S}}$ by averaging the scale-normalized shape of the N training faces in the cluster:

$$\bar{\mathcal{S}} = \left[\frac{1}{N} \sum_{n=1}^{N} \hat{\mathbf{a}}_n, \frac{1}{N} \sum_{n=1}^{N} \hat{\mathbf{b}}_n \right]. \quad (2)$$

A standard Delaunay triangulation $DT(\bar{\mathcal{S}}) = \{(k_i, l_i, m_i) \in [\![1, P]\!]^3\}_i$ is then computed from the P landmark locations in the average shape $\bar{\mathcal{S}}$. Each landmark triplet (k_i, l_i, m_i) defines a triangle in the standard image $\bar{\mathcal{S}}$ or in an arbitrary input image $\mathcal{S}$, see Fig. 4. The piecewise affine warping normalization then consists of warping the pixels in each triangle from the input image to the corresponding triangle in the standard image using the affine transform uniquely defined from the three pairs of matching vertices.

Sparse appearance modeling: The normalized face images described previously are rasterized to form signal vectors $\mathbf{z} \in \mathbb{R}^d$, with d the number of pixels. We will decompose the mean-removed signal vectors $\mathbf{y} = \mathbf{z} - \bar{\mathbf{z}}$ using

sparse coding. The mean vector $\bar{\mathbf{z}}$ is taken to be the average of all $\mathbf{z}$ vectors extracted form training images in the cluster of interest. In practice we will extract vectors $\mathbf{z}$ only from those positions $\mathcal{F}$ in the image that are expected to depict the face and not the background, see Fig. 6.

Using whole-image rasterization to produce signal vectors results in uncommonly large vector dimensions (e.g. $d = 10^4$ for $100x100$ images), and hence we will need to use an undercomplete dictionary matrix and learn it using stochastic gradient descent [3].

Sparse coding: Given a dictionary $\mathbf{D}$, we use a standard formulation of sparse coding

$$\mathbf{x}^{\circ}(\mathbf{y},\mathbf{D}) = \min_{\mathbf{x}} |\mathbf{y}-\mathbf{D}\mathbf{x}|_2^2 + \lambda|\mathbf{x}|_1 \quad (3)$$

relying on an ℓ_1 penalization $|\mathbf{x}|_1 = \sum_i |x_i|$ [1]. Given the decomposition $\mathbf{x}^{\circ}$ of the vector $\mathbf{y}$, an approximation $\hat{\mathbf{y}}$ of $\mathbf{y}$ can be obtained using $\hat{\mathbf{y}} = \mathbf{D}\mathbf{x}^{\circ}$.

Dictionary learning: In order to represent recurring spatial patterns, we will learn the dictionary matrix $\mathbf{D}$ required in (3) from the set of N training vectors $\{\mathbf{y}_n \in \mathbb{R}^d\}_{n=1}^N$ using the following objective:

$$\text{argmin}_{\mathbf{D},\{\mathbf{x}_n\}} \sum_{n=1}^{N} |\mathbf{y}_n - \mathbf{D}\mathbf{x}_n|_2^2 + \lambda\,|\mathbf{x}_n|_1,\ |\mathbf{d}_k|_2 \leq 1, \forall k. \quad (4)$$

Given the uncommonly large dimensionality of the signal space addressed by our face-inpainting approach, we use a learning method [27] based on per-atom block-coordinate descent using stochastic updates. At iteration $n < N$, the approach incorporates a randomly selected sample $\mathbf{y}_n$ into the solution for each column $\mathbf{d}_k$ of $\mathbf{D}$ by setting the gradient of (4) with respect to $\mathbf{d}_k$ to zero.

3.3. Face recovery

Once training has been performed off-line, we can recover missing/occluded pixels from any previously unseen face image. First, as before, we apply RCPR to estimate the face landmarks.

Note that RCPR is also capable of estimating whether landmarks are or not occluded (albeit with a low recall) if trained on a dataset containing occlusion ground-truth information, such as *COFW* [5]. When this is the case, RCPR will output the "degree of occlusion" of each landmark, adding a third component $\mathbf{o}$ to the shape parametrization: $\mathcal{S} = [\mathbf{a},\mathbf{b},\mathbf{o}]$ where $\mathbf{a},\mathbf{b},\mathbf{o} \in \mathbb{R}^P$ and $\mathbf{o} \in [0,1]$.

From this rough occlusion estimation, we can build an occlusion probability mask by performing Delaunay triangulation on the landmarks and setting each triangle's occlusion to the average of its anchor landmarks [2]. Fig. 5 shows example RCPR's results and the occlusion masks built for some occluded faces (from COFW test set).

[2]In fact we can re-use the triangulation computed for the face cluster-specific image warping, without further computation

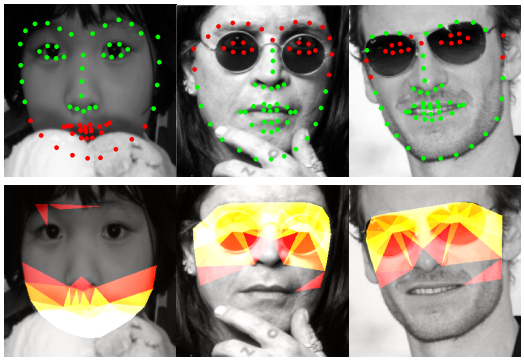


Figure 5: Example RCPR success cases. Top: landmark estimation results (landmarks with $o > .5$ are plotted in red to signal occlusion). Bottom: occlusion masks computed from landmarks (>intensity=>occlusion).

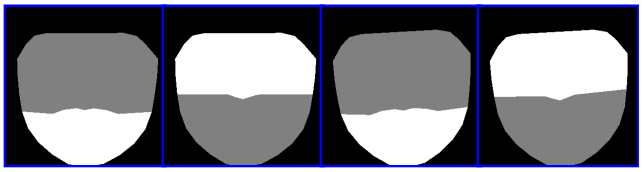


Figure 6: Examples of cluster-specific synthetic masks used in Figures 7, 8 and 9 .

Recovery using sparse coding Once the occlusion mask computed as described before, we conduct inpainting based on sparse coding as follows: sparse code of input image is computed using visible pixels only, and used to produce a complete reconstruction, including of occluded pixels. Let $\mathcal{A}$ represent the indices of the available pixels of $\mathbf{y}$. Letting $\mathbf{y}_{\mathcal{A}}$ (respectively $\mathbf{D}_{\mathcal{A}}$) denote the subvector (sub-matrix) obtained by retaining the coefficients (rows) at positions $\mathcal{A}$, an approximation of the whole vector $\mathbf{y}$ can be obtained from

$$\mathbf{D}\mathbf{x}^{\circ}(\mathbf{y}_{\mathcal{A}},\mathbf{D}_{\mathcal{A}}). \quad (5)$$

In Fig. 6 we depict four example configurations of $\mathcal{F}$ (in white and gray), background (black), available pixels (white) and missing pixels (gray).

4. Evaluation

4.1. Datasets

FaceScrub The *FaceScrub* dataset [32] consists of more than $100K$ images of 530 different actors and actresses, along with face bounding boxes. The dataset is provided as image urls, and currently only about 75% of the urls (76,800 images) are still valid and point to the correct data.

We split these available images into training and testing sets by holding out, as a testing set, all available images of 50 actors and 50 actresses.

This results in a training set composed of $29K/32K$ images of actresses/actors, respectively, and a testing set of

$7K/5K$ images. Of all these, for computational issues we use a random subset (roughly $8K/8K$ images for training and $2K/2K$ for testing). We further leave out some of the male actor's images as a validation set to tune parameters of our system and all other subspace learning techniques benchmarked.

In order to show some quantitative results (impossible on naturally occluded images), we use the test set together with synthetic occlusion masks (upper/lower part of the face) derived from the landmark detection associated to each cluster, see Fig. 6.

In Section 4.2 we explore the impact of parameters, while in Section 4.3 we compare our sparse coding approach against other popular techniques [12, 25, 7, 26]

COFW The *Caltech Occluded Faces in the Wild* (*COFW*) dataset [5] consists of 2K training and 500 test images with varied types and amounts of occlusions. Each image is made available together with the ground-truth 29 landmarks in the *LFPW* [2] format, with an extra flag that encodes whether each landmark is occluded or not.

We keep only the images from the test set, and use them to show-case results when coupling our method with a real-time occlusion detection system (provided by **RCPR**), see Sec.4.4. Please note that for these images we can only show qualitative results, since the occlusions are fully natural and therefore the "true" unoccluded face is unknown. Note also that when carrying out tests on COFW, we use clusters and dictionaries learned on the *FaceScrub* dataset, using COFW strictly as a testing set.

4.2. Parameter selection

We use the *FaceScrub* training set to build a set of clusters using a K-means algorithm in landmark space. Rather than specifying the number of clusters directly, we specify the minimum cluster size SZ, since the learning requires a minimum number of training examples. Since we use a hybrid approach that mixes clustering together with piecewise-affine warping, the canonical face landmark layout is computed on a per-cluster basis to be the mean landmark layout for all training faces in the cluster.

In Fig. 7, we evaluate the impact on our sparse coding approach of *(i)* the minimum cluster size SZ, *(ii)* the number of dictionary atoms N, and *(iii)* the testing and *(iv)* training sparsity levels. We plot Peak Signal to Noise Ratio (PSNR) $\log_{10}(\frac{255^2}{MSE})$, where the Mean Square Error (MSE) is computed over the pixels in the occluded region. The resulting PSNR value is computed over all clusters, and the average value is plotted along with the resulting standard deviation (as error bars).

Given that the face pose varies between landmark clusters, in these experiments we carry out parameter selection using cluster-dependent masks generated automatically

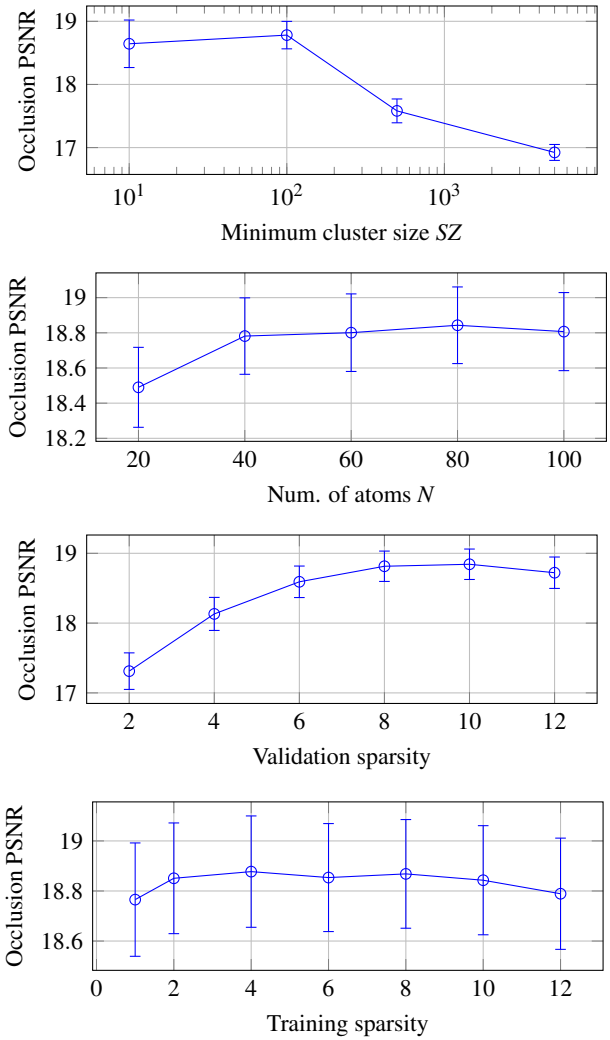


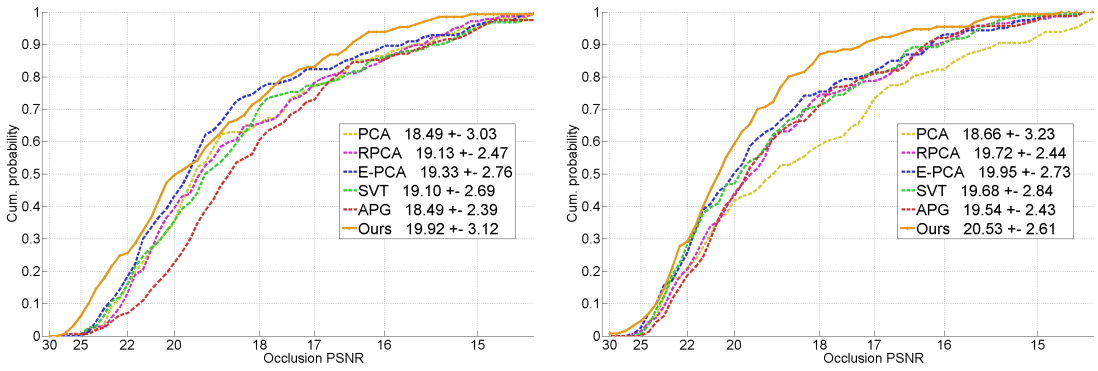
Figure 7: Occlusion PSNR over the validation set (male faces only) versus various system parameters.

from the mean landmark layout for the cluster so that the entire face is used as the signal vector support, and the top or bottom half of this support is used as an occluded region. We learn one dictionary in image space per landmark cluster using the FaceScrub training dataset, and cross validate against the FaceScrub validation set.

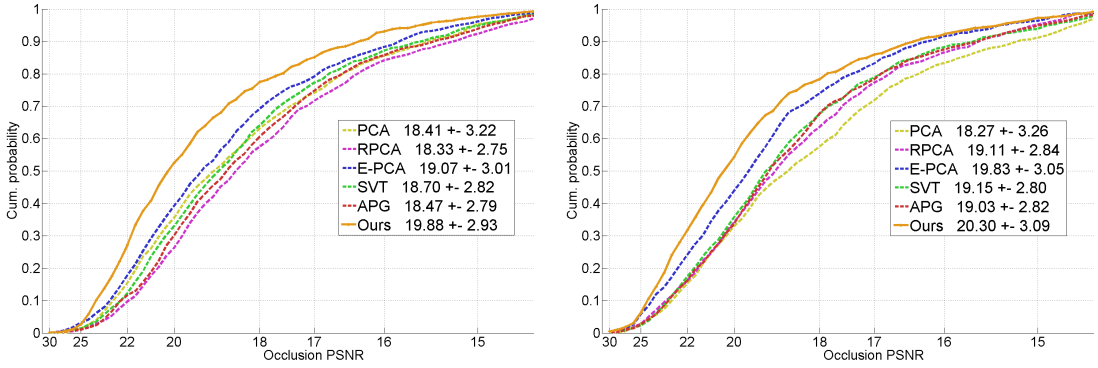
The parameter with most impact on performance is the cluster size SZ. The optimal value $SZ = 100$ reaches a compromise between having as many training faces as possible but without them being too different among themselves (the higher SZ the fewer clusters are found and therefore there is a higher intra-cluster variance). The clear drop in performance for high values of SZ justifies the need for the proposed face clustering.

4.3. Synthetic occlusion/missing data

We compare our sparse coding recovery method against several other popular techniques: (PCA [33], Robust-



(a) Upper mask, $SZ = 100$ (b) Lower mask, $SZ = 100$



(c) Upper mask, $SZ = 10$ (d) Lower mask, $SZ = 10$

Figure 8: Quantitative result comparison between our sparse coding approach and other techniques for different values of SZ and different occlusion masks. Each plot depicts the cumulative probability distribution of the PSNR between original and recovered face.

PCA [12], Aproximated Principal Gradient (APG) [25], Singular Value Thresholding (SVT) [7] and Euler-PCA [26]). All these techniques are applied within the context of our clustering-based framework, using strictly the same training/test faces than those used by our method. For fairness of comparison, we also used the validation set to tune the parameters of each one of these techniques, see Supp. Material for more info.

Fig. 8 shows the results for both $SZ = 10$ and $SZ = 100$ on both the upper and lower occlusion masks shown in Fig. 6. We compute occlusion PSNR values over all clusters and plot the probability distribution of each. Fig. 9 shows subjective comparisons for a selection of images from different clusters. Our approach outperforms all other techniques by 0.7 dB on average. It is also worth noting that the lower part of the face seems to be always easier to reconstruct than the upper part. Classic PCA seems to be the least competitive approach of all those benchmarked.

4.4. Real occlusions

We now show results of our method on *COFW* images displaying real face occlusions such as glasses, headwear, and hair. We apply **RCPR** to predict both the landmark's positions and occlusion, and from this information we derive an associated occlusion mask by weighing the components of a Delaunay triangulation of the landmarks based on the number of occluded vertices, as explained in Section 3.3. Since the resulting masks varies on a per-image basis, we train a new dictionary for each associated mask using the *FaceScrub* training examples of the associated cluster. We show some example results for faces where occlusion was correctly detected by **RCPR** in Fig. 10 (more available in Supp. Material).

Our method can output very realistic reconstructions even in these challenging conditions, under a variety of head poses, expressions and occlusions. Thanks to our proposed cluster-specific texture learning the reconstructions are realistic, leveraging the well known fact that natural human expressions are manifested in all parts of the face (*e.g.* the eyes take a particular form when one smiles).

5. Conclusion

We propose a novel method to recover lost pixels from a face image. Our method does not need identity, head pose or expression to be known a priori neither during training nor testing. During training, first we cluster faces based on

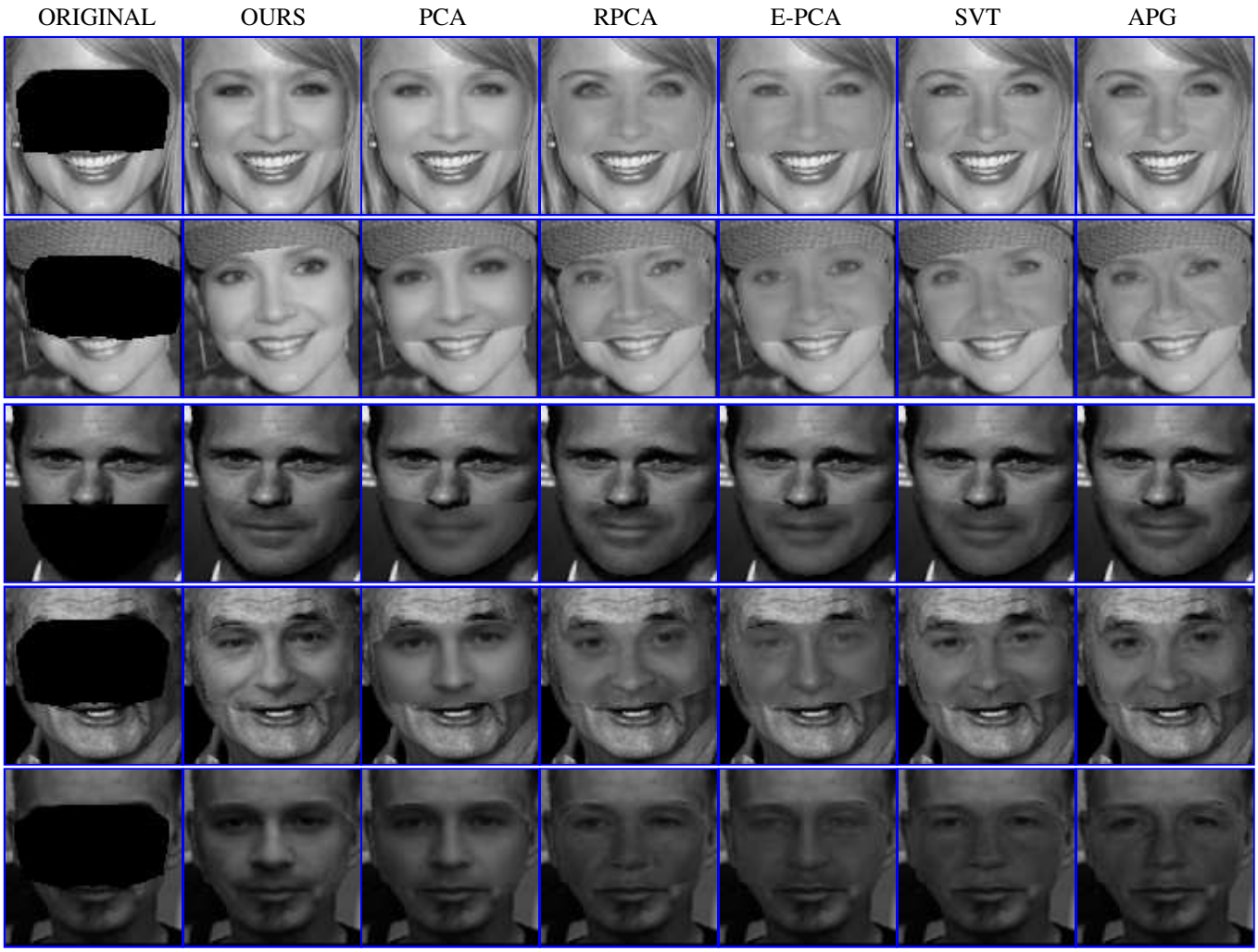


Figure 9: Reconstruction examples with $SZ = 100$ varying the reconstruction technique used after clustering. From left to right: original image with occluded region, Our Sparse coding approach, PCA, Robust-PCA [12], Euler-PCA [26], SVT [7] and APG [25].

Figure 10: Original image (*left*) and reconstruction using our proposed method (*right*) when using automatically detected occlusion masks.

their landmark's positions (obtained by an automatic face landmark estimator). Then, we model the face appearance for each group using sparse coding with cluster-specific dictionaries. At test time, given a face to recover, we find its belonging cluster and occluded area and restore missing pixels by applying the group-specific sparse appearance representation learned during training.

Systems that carry out automatic occlusion detection and reconstruction have important applications, for example, in augmented reality settings, as well as a visual aid for people with conditions, such as prosopagnosia, preventing them from easily recognizing faces (conditions exacerbated by the presence of occlusions). The results illustrated in Fig. 10 suggest that building such a system is indeed possible.

References

[1] F. Bach. Sparse methods for machine learning theory and algorithms. Technical report, 2010. 5

[2] P. Belhumeur, D. Jacobs, D. Kriegman, and N. Kumar. Localizing parts of faces using a concensus of exemplars. In *CVPR*, 2011. 6

[3] L. Bottou. Large-scale machine learning with stochastic gradient descent. In *COMPSTAT*. Springer, 2010. 5

[4] O. Bryt and M. Elad. Compression of facial images using the k-svd algorithm. *JVCI*, 19(4):270–282, May 2008. 3

[5] X. P. Burgos-Artizzu, P. Perona, and P. Dollár. Robust face landmark estimation under occlusion. In *ICCV*, 2013. 1, 2, 3, 5, 6

[6] X. P. Burgos-Artizzu, M. R. Ronchi, and P. Perona. Distance estimation of an unknown person from a portrait. In *ECCV*, pages 313–327. Springer, 2014. 3

[7] J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM J. on Optimization*, 20(4):1956–1982, 2010. 1, 2, 6, 7, 8

[8] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In *CVPR*, 2012. 3

[9] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *PAMI*, 23(6):681–685, 2001. 2, 3

[10] T. Cootes and C. Taylor. Active shape models. In *BMVC*, 1992. 3

[11] A. Criminisi, P. Pérez, and K. Toyama. Region filling and object removal by exemplar-based image inpainting. *IP*, 13(9):1200–1212, 2004. 2

[12] F. De la Torre and M. J. Black. Robust principal component analysis for computer vision. In *ICCV*, 2001. 1, 2, 6, 7, 8

[13] D.Lin and X. Tang. Quality-driven face occlusion detection and recovery. In *CVPR*, 2007. 1, 2

[14] P. Ekman and W. Friesen. *Facial action coding system*. 1977. 1

[15] M. Everingham, J. Sivic, and A. Zisserman. Hello! My name is... Buffy - automatic naming of characters in tv video. In *BMVC*, 2006. 1

[16] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 32(9):1627–1645, 2010. 3

[17] G. Ghiasi and C. Fowlkes. Occlusion coherence: Localizing occluded faces with a hierarchcial deformable part model. In *CVPR*, 2014. 2, 3

[18] J. Hays and A. A. Efros. Scene completion using millions of photographs. 2007. 2

[19] T. Hosoi, S. Nagashima, K. Kobayashi, K. Ito, and T. Aoki. Restoring occluded regions using fw-pca for face recognition. In *CVPR-Workshops*, 2012. 1, 2

[20] D. Huang and F. De La Torre. Facial action transfer with personalized bilinear regression. In *ECCV*, pages 144–158. Springer, 2012. 2

[21] B.-W. Hwang and S.-W. Lee. Reconstruction of partially damaged face images based on a morphable face model. *PAMI*, 25(3):365–372, 2003. 1, 2

[22] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for sparse hierarchical dictionary learning. 2010. 3

[23] M. Kass, A. Witkin, and D. Terzopoulos. Snakes:active contour models. *IJCV*, 1(4):321–331, 1988. 3

[24] J. Lee and N. Kwak. Detection and recovery of occluded face images based on correlation between pixels. In *ICPRAM*, 2012. 1

[25] Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, and Y. Ma. Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. In *Workshop on Comp. Adv. in Multi-Sensor Adapt. Proc.*, 2009. 1, 2, 6, 7, 8

[26] S. Liwicki, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. Euler principal component analysis. *IJCV*, 101(3):498–518, 2013. 1, 2, 6, 7, 8

[27] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *ICML*, 2009. 5

[28] A. Martinez and S. Du. A model of the perception of facial expressions of emotion by humans: Research overview and perspectives. *JMLR*, 13:1589–1608, 2012. 1

[29] A. M. Martinez and R. Benavente. The ar face database. Technical report, Purdue University, 2000. 2

[30] I. Matthews and S. Baker. Active appearance models revisited. *IJCV*, 60:135–164, 2004. 3

[31] Z. Mo, J. Lewis, and U. Neumann. Face inpainting with local linear representations. In *BMVC*, 2004. 1, 2

[32] H. Ng and S. Winkler. A data-driven approach to cleaning large face datasets. In *ICIP*, 2014. 1, 2, 5

[33] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 1901. 2, 6

[34] P. Phillips, H. Moon, P. Rauss, and S. Rizvi. The feret evaluation methodology for face recognition algorithms. *PAMI*, 2000. 2

[35] S. Ren, X. Cao, Y. Wei, and J. Sun. Face alignment at 3000 fps via regressing local binary features. In *CVPR*, 2014. 3

[36] R. Rubinstein, S. Member, M. Zibulevsky, M. Elad, and S. Member. Double sparsity : Learning sparse dictionaries for sparse signal approximation. *SP*, 58(3):1553–1564, 2010. 3

[37] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *ICCV-Workshop*, 2013. 3

[38] D. Vlasic, M. Brand, H. Pfister, and J. Popović. Face transfer with multilinear models. In *TOG*, volume 24(3), pages 426–433. ACM, 2005. 2

[39] S. J. Wright, R. D. Nowak, S. Member, and M. A. T. Figueiredo. Sparse reconstruction by separable approximation. *IP*, 57(7):2479–2493, 2009. 3

[40] J. Yang, H. Tang, Y. Ma, and T. Huang. Face hallucination via sparse coding. In *ICIP*, 2008. 3

[41] D. Yu and S. T. Using targeted statistics for face regeneration. In *FG*, 2008. 1, 2

[42] A. L. Yuille, P. Hallinan, and D. S. Cohen. Feature extraction from faces using deformable templates. *IJCV*, 8(2):99–111, 1992. 3

[43] J. Zepeda, C. Guillemot, and E. Kijak. Image compression using the iteration-tuned and aligned dictionary. In *ICASSP*, 2011. 3