IMAGE INTERPOLATION USING GAUSSIAN MIXTURE MODELS WITH SPATIALLY CONSTRAINED PATCH CLUSTERING

Milad Niknejad[1]*, Hossein Rabbani*[2] *, Massoud Babaie-Zadeh*[3]*, Christian Jutten*[4]

[1]Islamic Azad University, Majlesi Branch, Iran
[2] Department of Biomedical Engineering, Isfahan University of Medical Sciences, Iran
[3]Sharif University of Technology, Department of Electrical Engineering, Tehran, Iran
[4]GIPSA-lab, Grenoble, and Institut Universitaire de France, France

ABSTRACT

In this paper we address the problem of image interpolation using Gaussian Mixture Models (GMM) as a prior. Previous methods of image restoration with GMM have not considered spatial (geometric) distance between patches in clustering, failing to fully exploit the coherency of nearby patches. The GMM framework in our method for image interpolation is based on the assumption that the accumulation of similar patches in a neighborhood are derived from a multivariate Gaussian probability distribution with a specific covariance and mean. An Expectation Maximization-like (EM-like) algorithm is used in order to determine patches in a cluster and restore them. The results show that our image interpolation method outperforms previous state-of-the-art methods with an acceptable bound.

Index Terms— Image restoration, interpolation, Gaussian mixture models, neighborhood clustering, continuation

1. INTRODUCTION

Generally, in the image restoration tasks the degraded image $\mathbf{y}$ (in vectorized form) can be mathematically modeled by

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v} \tag{1}$$

where $\mathbf{x}$ is the clean image in the vectorized form, $\mathbf{H}$ is a non-invertible linear operator and $\mathbf{v}$ is the vector of independent Gaussian noise with known variance σ^2. However, in this paper, we address the image interpolation case in which the observation is noise free, i.e. $\mathbf{v}$ is zero, and the degrading operator is a subsampling matrix, i.e. $\mathbf{H}$ is diagonal matrix with one or zero diagonal entries corresponding to the existing or missing pixels, respectively. We consider both randomly and uniformly observed pixels which the latter case is equivalent to image zooming. Our method, similar to the successful image restoration methods in [1] , [2] and [3], is a patch-based method in which the image is divided into local $\sqrt{n} \times \sqrt{n}$ sized overlapping patches. Each patch, denoted by $\mathbf{y}_i \in \mathbb{R}^n$ in vectorized form, can be modeled by $\mathbf{y}_i = \mathbf{H}_i\mathbf{x}_i$ where $\mathbf{H}_i$ is degrading matrix corresponding to the underlying clean patch $\mathbf{x}_i$.

Some image interpolation methods are based on the assumption of sparsity in the representations [1, 4]. However, the relatively strict conditions for obtaining the exact sparse representations [5], especially in the cases like interpolation where the degrading operator $\mathbf{H}$ is not the identity matrix (i.e. the problem is not denoising), is a drawback. The main reason is the deformation of the dictionary structure by the matrix $\mathbf{H}$, and consequently increasing the mutual coherence of dictionary which makes meeting these conditions harder [3, 6]. To overcome this drawback, methods that are based on Gaussian Mixture Models (GMM) such as Piecewise Linear Estimation (PLE) [3] and Expected Patch Log Likelihood (EPLL) [7] have been proposed. As shown in these methods, the GMM prior for image patches leads to linear estimations in solving (1) which avoids the strict sparse coding recovery conditions. However, the method of clustering used in the GMM methods does not consider the geometric distance between patches. In some successful image denoising methods like nonlocal-means [8] and bilateral filtering [9], geometrical distance is considered by averaging pixels with the weights inversely proportional to distance between pixels or patches, to exploit coherency of nearby pixels or patches.

In this paper, we introduce the idea of applying GMM to nearby patches in order to interpolate images with partially observed pixels. We propose a model that uses a same multivariate Gaussian probability distribution for similar image patches in a neighborhood. In other words, we assume that k-Nearest-Neighbor (kNN) patches with respect to an exemplar patch are derived from a multivariate Gaussian probability distribution with a specific covariance matrix and mean vector. An Expectation Maximization-like (EM-like) approach [10] is used to obtain accurate clustering of patches and estimating underlying covariances and means in the mixture distributions. In the averaging of the overlapped patches to con-

This work was partially funded by European project 2012-ERC-AdG-320684 CHESS, and by Iran National Science Foundation (INSF) under contract number 91004600.

struct the whole image, we consider assigning the averaging weights proportional to similarity to Gaussian distributions, i.e. patches that are more likely to be generated from the estimated Gaussian distribution of their allocated clusters, benefit from higher weights. Similar to the recently proposed methods in [1] and [11] for image interpolation, a continuation approach, which reduces the regularization parameter along iterations of minimizing the cost function to avoid local minima, is used. The results show that our method outperforms the state of the art image interpolation methods in recovering randomly and uniformly sampled images.

In the following sections, first, the previous methods which use GMM for image restoration are briefly described. Then, section 3 is devoted to the description of our proposed method for image interpolation. In section 4, we compare our method with the state of the art methods of image interpolation.

2. GLOBAL IMAGE RESTORATION METHODS USING GMM

PLE and EPLL methods, which use GMM for patches in the image, are very similar with minor differences in the initialization and computing the aggregation weights. We call the two mentioned methods global GMM methods in some parts of this paper, since no spatial (geometric) distance between patches are considered and patches from very different parts of image could be clustered into one group. Generally, these methods assume that every patch in the image is derived from a multivariate Gaussian probability distribution $\mathcal{N}(\mathbf{\Sigma}_m, \mu_m)$ which is parameterized by the covariance matrix $\mathbf{\Sigma}_m$ and the mean vector μ_m. They also assume that there are M finite numbers of Gaussian distributions i.e. $\{\mathcal{N}(\mathbf{\Sigma}_m, \mu_m)\}_{1<m<M}$ in the image. So each patch $\mathbf{x}_i$ is independently drawn from one of these finite number of Gaussians with the probability of

$$P(\mathbf{x}_i) = \frac{1}{(2\pi)^{\frac{n}{2}} |\mathbf{\Sigma}_m|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}_i - \mu_m)^T \mathbf{\Sigma}_m^{-1} (\mathbf{x}_i - \mu_m)} \quad (2)$$

Maximizing the above probability distribution for all patches with the assumption of finite Gaussian distributions in the whole image are obtained by the following steps in the global GMM methods with some initial Gaussian distributions:

- The Gaussian probability that most likely generates each patch is determined from $\{\mathcal{N}(\mathbf{\Sigma}_m, \mu_m)\}_{1<m<M}$. This can be seen as clustering of patches and the similarity is measured by (2).
- The estimation of the covariance matrix and the mean vector, $(\mathbf{\Sigma}_m, \mu_m)$, for each $1 < m < M$, are updated based on the patches and the corresponding clusters.
- The restoration of each patch is obtained based on its allocated Gaussian distribution.

In the first step of the above Global GMM algorithm, each patch is assigned to the finite number of Gaussians in the whole image and the geometric distance between patches is not considered. This global clustering prevents from fully exploiting the coherency of nearby patches in the image. Using coherency of nearby patches has been considered in image restoration algorithms, especially in the denoising case. Nonlocal means [8] and bilateral filtering [9], which rely on averaging pixels, consider the coherency of nearby patches through setting averaging weights inversely proportional to geometrical distance between pixels or patches. Also, some recent image denoising methods such as Block Matching 3D (BM3D) [2] constrain grouping of similar patches in a window with finite-size and then collaboratively denoise them.

A question that can be posed here is how to efficiently develop an image interpolation method benefiting from the constraint discussed above to improve the restoration task using GMM. One may propose to constrain GMM into a finite-sized window in different parts of image, and for each window, use a global GMM method. At the first glance, this suggestion seems convincing, but it suffers from a problem of appearing block artifact. Constraining denoising in blocks of windows, for example 32×32 sized window, leads to block artifacts in boundaries of windows in the restored image. Even if the overlapping is considered, these artifacts appear, unless they are consider in every 1 or 2 pixel jump sizes which is not applicable due to high redundancy and high computational complexity.

3. OUR PROPOSED METHOD

In order to apply a plausible spatial constraint on the image, our proposed method defines a new GMM framework for the image interpolation which states that in the image, similar patches in a neighborhood are derived from a same multivariate Gaussian probability distribution with a specific mean and covariance.

Our method, similar to BM3D denoising [2] and BM3D-based interpolation [1] methods, collects exemplar patches chosen uniformly with an appropriate jump size in the row and the column of the whole image and groups similar patches in the neighborhood of each exemplar patch. The neighborhood for an exemplar patch is defined as $N \times N$ sized window around that patch. The r^{th} region related to the r^{th} exemplar patch is defined as kNN patches with respect to that patch. An important issue here is finding Nearest Neighbor patches with respect to the exemplar patches in the image, while the observed patches are severely degraded. In our method, in order to determine kNN patches while having degraded observations, similar to [12], we use an EM-like approach in which kNN patches are treated as missing variables. In the E-like step kNN patches are determined, and in the M-like step the image is restored by assumption of multivariate Gaussian distributions for the image patches. Let $\{\mathbf{x}_r\}_{r=1,R}$ denotes the collection of exemplar patches and let $\{\mu_r\}_{r=1,R}$ and $\{\mathbf{\Sigma}_r\}_{r=1,R}$ denote the corresponding mean

vectors and covariance matrices, respectively. After initialization of $\hat{\mathbf{x}} = \mathbf{y}$, our method iteratively implements the following steps:

E-like step: As we assumed assigning kNN patches to the exemplar patch $\mathbf{x}_r$ as hidden variables of EM-like algorithm, in this step for all exemplar patches we collect k patches in the neighborhood of the exemplar patch that have the minimal dissimilarity d to the exemplar patch. The dissimilarity is simply measured by l_2-norm metric i.e. $d = \|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_r\|_2^2$ for all $\hat{\mathbf{x}}_i$'s in the neighborhood of $\hat{\mathbf{x}}_r$. Note that $\hat{\mathbf{x}}_i$'s and $\hat{\mathbf{x}}_r$ are estimated patches obtained from the previous M-like step or the initialization.

M-like step: This step is mainly comprised of the process of restoring the image using the clusters obtained from the previous E-like step. To achieve this, the parameters of Gaussian distributions for each group of patches should be estimated first. In order to obtain each cluster covariance matrix and mean vector, $(\hat{\mu}_r, \hat{\mathbf{\Sigma}}_r)$, like PLE, Maximum Likelihood (ML) estimate is employed which leads to the sample covariance matrix and the sample mean vector i.e. $\hat{\mu}_r = \frac{1}{k}\sum_{i\in\mathcal{S}_r}\hat{\mathbf{x}}_i, \hat{\mathbf{\Sigma}}_r = \frac{1}{k}\sum_{i\in\mathcal{S}_r}(\hat{\mathbf{x}}_i - \hat{\mu}_r)(\hat{\mathbf{x}}_i - \hat{\mu}_r)^T$ where $\mathcal{S}_r$ is the set of kNN patches with respect to the r^{th} exemplar patch. Having estimated the parameters of Gaussian distributions, in order to obtain the restored patch $\hat{\mathbf{x}}_i$ in the r^{th} region from the corresponding noisy observed patch $\mathbf{y}_i$, similar to Global GMM methods, log a-posteriori probability of the form $\log p(\mathbf{x}|\mathbf{y}_i, \hat{\mathbf{\Sigma}}_r, \hat{\mu}_r)$ is maximized i.e.

$$\begin{aligned}\hat{\mathbf{x}}_i &= \underset{\mathbf{x}}{\operatorname{argmax}} \log p(\mathbf{x}|\mathbf{y}_i, \hat{\mu}_r, \hat{\mathbf{\Sigma}}_r) \\ &= \underset{\mathbf{x}}{\operatorname{argmax}} \log p(\mathbf{y}_i|\mathbf{x}, \hat{\mu}_r, \hat{\mathbf{\Sigma}}_r) p(\mathbf{x}|\hat{\mathbf{\Sigma}}_r, \hat{\mu}_r) \\ &= \underset{\mathbf{x}}{\operatorname{argmin}} \|\mathbf{y}_i - \mathbf{H}_i\mathbf{x}\|_2^2 + \sigma^2(\mathbf{x} - \hat{\mu}_r)^T\hat{\mathbf{\Sigma}}_r^{-1}(\mathbf{x} - \hat{\mu}_r) \quad (3)\end{aligned}$$

where the second equality is obtained by the Bayes rule and the third equality is derived from the assumptions of $\mathbf{x}_i \sim \mathcal{N}(\hat{\mu}_r, \hat{\mathbf{\Sigma}}_r)$ and $\mathbf{v}_i \sim \mathcal{N}(0, \sigma^2\mathbf{I})$. Note that in the noiseless interpolation case, the Gaussian noise variance in (3) can be considered as a small value [3] (for example this value is set to 3 in PLE to correspond to typical noise level existing in the images). The problem in (3) can also be seen as the regularized form of the patch restoration problem in which σ^2 is the regularization parameter. So for simplicity of the notations, we replace σ^2 by λ in the subsequent formulas. The convex optimization problem in (3) is solved by setting its derivative to zero which leads to the Wiener filter of the form

$$\hat{\mathbf{x}}_i = (\mathbf{H}_i^T\mathbf{H}_i + \lambda\mathbf{\Sigma}_r^{-1})^{-1}(\mathbf{H}_i^T\mathbf{y}_i + \lambda\mathbf{\Sigma}_r^{-1}\mu_r). \quad (4)$$

The restored patches, obtained by the above formula, should be returned to their original positions to construct the whole image. Using the proper weights in averaging the patches improves the performance of the restoration method [2]. For example in BM3D, these weighs are set proportional to the number of non-zeros in the representations of the patches. Due to using multivariate GMM in our method, we propose to obtain averaging weights by the degree of similarity to Gaussian structures. By this, we mean that the patches which have higher probability of occurrence in the estimated Gaussian distributions from which they are derived, average with higher weights. So, in order to determine these weights, we use the help of Probability Density Function (PDF) of patch given Gaussian parameters, mentioned in (2), in which $(\mu_r, \mathbf{\Sigma}_r)$ is fixed and is equal to the estimated parameters $(\hat{\mu}_r, \hat{\mathbf{\Sigma}}_r)$ at each iteration. Using exactly (2) for obtaining weights may not lead to optimum interpolation performance and can be scaled properly to have a better performance. So in order to have a proper scale of (2) for averaging weights, we use

$$w_{(i,r)} = \frac{1}{\sqrt{|\hat{\mathbf{\Sigma}}_r|}} e^{-\frac{\gamma}{2}(\hat{\mathbf{x}}_i - \hat{\mu}_r)^T\hat{\mathbf{\Sigma}}_r^{-1}(\hat{\mathbf{x}}_i - \hat{\mu}_r)} \quad (5)$$

where $w_{(i,r)}$ relatively measures the mentioned probability. We empirically found the formula is efficient in the image interpolation. In comparison to (2), in the above formula we omit the constants and add γ which is an appropriate scaling constant. Applying a constant like γ in the power of exponential to achieve a proper scale has been shown to be effective in image restoration tasks, for example in computing averaging weights for pixels in the bilateral filter [9], or in some spatial domain image restoration kernels computed based on similarity of pixels [13]. However, we use this parameter in the power of exponential to distinguish between averaging weights assigned to patches.

The overall problem of clustering and finding Gaussian distributions for patches is non-convex, and the solution may be trapped into local minima [3]. The continuation method, which starts from a high regularization parameter and gradually decreases this parameter along iteration, has been successfully used in the image interpolation methods in [14] and [15], to avoid local minima. In many methods, using this approach is done heuristically by a linear or exponential decay. Starting from a high value of the regularization parameter, in our method the exponential decay is used, the same as the one successfully used for image interpolation in [15], i.e. at the n^{th} iteration this parameter is obtained by $\lambda_n = (1 - \epsilon)\lambda_{n-1}$ where ϵ is a small constant.

Straightforward implementation of our algorithm described in this section, needs high memory usage for storing numerous covariance matrices. However, our method similar to BM3D benefits from the capability of online implementation by allocating two buffers for the weighted restored patches and the aggregation weights. The patches that are similar to an exemplar patch in a finite-sized window are grouped together, the mean and the covariance are estimated directly from the grouped patches, and after the restoration, the restored patches with the obtained weights are accumulated in the buffers. This procedure repeats for all exemplar patches respectively at each iteration. The restored image is obtained by element wise division of two buffers.

Table 1. Image interpolation results PSNR (dB) values for competing methods with different percentages of randomly observed pixels in some benchmark images.

	% of observed pixels	BP [16]	KR [17]	PLE [3]	This work
Barbara	80%	40.76	37.81	43.85	**43.92**
	50%	33.17	27.98	37.03	**37.48**
	30%	27.52	24.00	32.73	**33.68**
Lena	80%	41.27	41.68	43.38	**43.60**
	50%	36.94	36.77	37.78	**37.98**
	30%	33.31	33.55	34.37	**34.53**
House	80%	43.03	42.57	44.77	**45.21**
	50%	38.02	36.82	38.97	**39.43**
	30%	33.14	33.62	34.88	**35.16**
Boats	80%	39.50	37.91	40.49	**40.70**
	50%	33.78	32.70	34.36	**34.58**
	30%	30.00	29.28	30.77	**30.81**

4. EXPERIMENTAL RESULTS

We evaluate the performance of our method with other state of the art methods including global GMM methods and methods based on sparsity in the image representations. Although, as mentioned, PLE and EPLL are very similar, PLE method generally performs better than EPLL, and we choose PLE among global GMM methods in this section for comparison. To the best of our knowledge, PLE outperforms other methods proposed so far for image interpolation in the open literature. In our implementation, $k = 25$ patches were gathered in a neighborhood of 32×32 sized window. For obtaining the regularization parameter along iterations, the initialization value and ϵ are set to 100 and 0.1, respectively. The exemplar patches were chosen every 5 pixels along both row and column directions of the image. The value of γ is 0.01 in (5) for computing aggregation weights. We empirically found that these parameters lead to acceptable performance in the image interpolation. The results are reported for 10 iterations of our algorithm, whose MATLAB implementation takes about 140 seconds to restore a 256×256 sized image on a 2.8 GHz Intel Core i7 CPU.

In Table 1, PSNR results of our proposed method are compared to the recent state-of-the-art methods of image interpolation, such as Beta Process (BP) [16], Kernel Regression (KR) [17] and PLE [3], for different percentages of randomly observed pixels. It can be seen that our method outperforms all state of the art methods including PLE in all percentages.

Figure 1 shows examples of image interpolation of some image fragments for 30% of observed pixels. This feagments are illusterated to focus on comparing the interpolation of both smooth and textured fragments of images. It can be seen that our method outperforms other method in both textured and smooth regions in terms of PSNR.

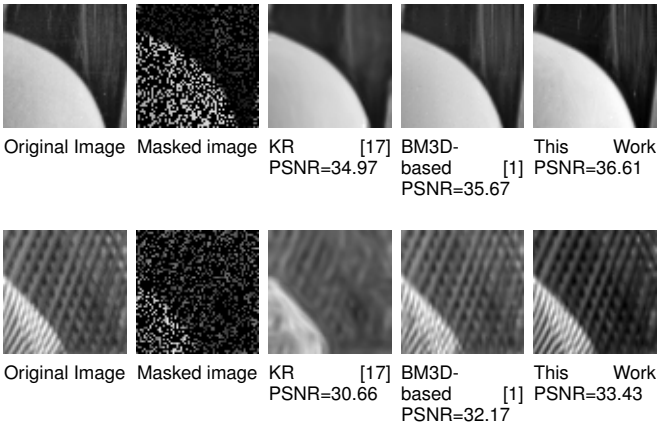

Fig. 1. Examples of image fragments interpolation from 30% of available data with different methods.

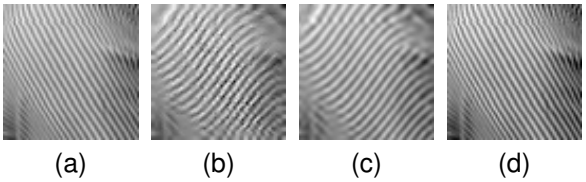

Fig. 2. Ability to recover true textures in uniformly sampled images in our method in comparison with other methods (zooming factor $= 2$):(a) Original image; (b) NEDI [4]; (c) BM3d-based [1]; (d) This work.

A special case of image interpolation is zooming which can be perceived as the interpolation of uniformly sampled images. However zooming is a more challenging task than interpolation from randomly observed pixels. Due to the regular sampling, many algorithms proposed for the interpolation fail to recover true underlying textures in images. As discussed in [1], more random sampling achieves dramatically better results. We found that our algorithm is highly robust to the recovery error caused by the uniform sampling. In Fig. 2, examples of image interpolation with some recent methods for a textured image fragment are illustrated. It can be seen that our method is noticeably successful to find true textures.

5. CONCLUSION

In this paper we proposed a method for image interpolation based on GMM in which the clustered patches are constrained into finite-sized windows. We also obtained the aggregation weights proportional to the probability of the restored patch given the estimated Gaussian parameters. The results show that our method outperforms the previous state of the art methods of the image interpolation in terms of PSNR.

6. REFERENCES

[1] X. Li, "Patch-based nonlocal image interpolation: algorithms and applications.," in *Local and Non-Local Approximation in Image Processing*, 2008.

[2] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3d transform-domain," *IEEE Trans. on Image Processing*, vol. 6, pp. 2080–2095, Aug. 2007.

[3] G. Yu, G. Sapiro, and S. Mallat, "Solving inverse problems with piecewise linear estimators: From gaussian mixture models to structured sparsity.," *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2481–2499, 2012.

[4] X. Li and M. T. Orchard, "New edge-directed interpolation.," *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1521–1527, 2001.

[5] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition.," *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 2845–2862, 2001.

[6] W. Dong, L. Zhang, R. Lukac, and G. Shi, "Sparse representation based image interpolation with nonlocal autoregressive modeling.," *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1382–1394, 2013.

[7] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration.," in *ICCV*, 2011, pp. 479–486.

[8] A. Buades, B. Coll, and M. Morel, "A non-local algorithm for image denoising," in *CVPR*, June 2005, pp. 60–65.

[9] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images.," in *ICCV*, 1998, pp. 839–846.

[10] Radford M Neal and Geoffrey E Hinton, "A view of the em algorithm that justifies incremental, sparse, and other variants," in *Learning in graphical models*, pp. 355–368. Springer, 1998.

[11] K. O. Egiazarian, A. Foi, and V. Katkovnik, "Compressed sensing image reconstruction via recursive spatially adaptive filtering.," in *ICIP*, San Antonio, TX, 2007, pp. 549–552.

[12] X. Li, "Exemplar-based em-like image denoising via manifold reconstruction.," in *ICIP*, 2010, pp. 73–76.

[13] P. Chatterjee and P. Milsnifar, "Nonlocally centralized sparse representation for image restoration," *IEEE Trans. on Image Processing*, vol. 18, pp. 1438 – 1451, 2009.

[14] X. Li, "Image recovery via hybrid sparse representations: A deterministic annealing approach," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 5, pp. 953–962, 2011.

[15] Luis Mancera and Javier Portilla, "Non-convex sparse optimization through deterministic annealing and applications.," in *ICIP*. 2008, pp. 917–920, IEEE.

[16] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, David B. Dunson, G. Sapiro, and L. Carin, "Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images.," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 130–144, 2012.

[17] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction.," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, 2007.