ADAPTIVE PATCH-BASED IMAGE DENOISING BY EM-ADAPTATION

Stanley H. Chan [1]*, Enming Luo*[2]*, and Truong Q. Nguyen*[2]

[1]School of ECE and Dept of Statistics, Purdue University, West Lafayette, IN 47907.
[2]University of California, San Diego, Dept of ECE, 9500 Gilman Drive, La Jolla, CA 92093.

ABSTRACT

Effective image prior is a key factor for successful image denoising. Existing learning-based priors require a large collection of images for training. Besides being computationally expensive, these training images do not necessarily correspond to the noisy image of interest. In this paper, we propose an adaptive learning procedure for learning image patch priors. The new algorithm, called the EM adaptation, maps a generic prior to a targeted image to create a specific prior. EM adaptation requires significantly less amount of training data compared to the standard EM, and can be applied to pre-filtered images in the absence of clean databases. Experimental results show that the adapted prior is consistently better than the originally un-adapted prior, and has superior performance than some state-of-the-art algorithms such as EPLL and BM3D.

***Index Terms*—** Image denoising, Gaussian Mixture models, Expectation-Maximization, Expected Patch Log-Likelihood, EM-adaptation, BM3D

1. INTRODUCTION

1.1. Overview

We consider the classical image denoising problem of recovering an image from a noisy observation, where the noise is i.i.d. Gaussian. Mathematically, the problem is to solve an inverse problem by finding $\boldsymbol{x} \in \mathbb{R}^n$ from

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{\varepsilon}, \tag{1}$$

where $\boldsymbol{y} \in \mathbb{R}^n$ is the observed image, and $\boldsymbol{\varepsilon} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I})$ is a Gaussian noise vector of variance σ^2.

Despite the long history of the problem, image denoising remains an active research area in which new methods are rapidly developed. In this paper, we focus on a class of methods known as the *patch-based image denoising* algorithms [1–10]. The common principle behind these methods is to partition a noisy image into overlapping patches. Each patch is then denoised and combined to reconstruct the image.

Patch-based image denoising algorithms rely heavily on the prior models they use. These priors, in general, are learned from either the single image (a.k.a. *internal* statistics [11]) or a database of image patches (a.k.a. *external* statistics [12, 13]). For example, non-local means (NLM) [1] and BM3D [3] are internal methods as they group similar patches from the noisy image and compute the denoised results using linear (and nonlinear) operations of these patches; KSVD [14] is an external method as it trains an over-complete dictionary using patches from a database and denoises the image through sparse optimization; EPLL [15] is another external method as it learns a Gaussian mixture model from millions of patches and denoises the image through a maximum-a-posteriori optimization. To date, there is still no consensus about which of the internal or external methods is better. However, it is generally known that internal methods often fail for rare patches, whereas external methods are computationally expensive.

Contact author: S. Chan, E-mail: stanleychan@purdue.edu. The work of E. Luo and T. Q. Nguyen is supported, in part, by a NSF grant CCF-1065305.

1.2. Related Works

In response to the limitations of internal and external methods, we propose to take an *adaptive* approach by adapting external priors to internal priors. Adaptive methods are known techniques in signal processing [16], but their usage in patch-based image denoising is relative new. In [17], Luo *et al.* explored adaptive methods to optimally design the search range of NLM for multiview image denoising. Later, the authors proposed a near-optimal external method by adapting the noisy image to a targeted database [18, 19]. They demonstrated superior image denoising results for a number of scenarios such as face and text images.

Besides adaptive methods, there are also *fusion* methods aiming to combine internal and external statistics. In [12], Mosseri *et al.* proposed a patch signal to noise ratio as a quantitative metric to decide whether a patch should be denoised internally or externally. A similar concept was proposed by Burger *et al.* [13], where they applied a neural network approach to learn optimal weights to combine internal and external denoising results. There are also methods trying to fuse the internal and external denoising results in the frequency domain [20]. A recent work by Sulam and Elad [21] attempted the same problem by combining EPLL with KSVD.

1.3. Contributions and Organization

The key contribution of this paper is an *EM-adaptation* algorithm. Different from fusion methods which only combine internal and external methods via a weighted average, we learn a single unified patch prior through an adaptive process which takes an external prior (called the generic prior) and maps it to a targeted noisy image to generate a specific prior.

The advantages of the proposed method are three-fold. First, by adapting the generic prior to the specific image, we avoid the ad-hoc procedure of finding the fusion weights. Second, our method can be applied even if there are very few training samples. This makes the algorithm more favorable than other learning methods such as EPLL [15] and PLE [22] which fail to learn from a single image. Third, the proposed method has a very low computational complexity. It is equivalent to one standard EM iteration only.

The rest of the paper is organized as follows. After introducing some mathematical preliminaries in Section 2, we present our proposed method in Section 3. Experimental results are shown in Section 4, and concluding remarks are given in Section 5.

2. MATHEMATICAL PRELIMINARIES

2.1. Patch Prior with Gaussian Mixtures

We formulate the denoising problem as a maximum-a-posteriori (MAP) estimation problem and we aim to solve the optimization

$$\begin{aligned}\underset{\boldsymbol{x}}{\operatorname{argmax}}\ f(\boldsymbol{y}|\boldsymbol{x})f(\boldsymbol{x}) &= \underset{\boldsymbol{x}}{\operatorname{argmin}}\ -\log f(\boldsymbol{y}|\boldsymbol{x}) - \log f(\boldsymbol{x}) \\ &= \underset{\boldsymbol{x}}{\operatorname{argmin}}\ \frac{\sigma^{-2}}{2}\|\boldsymbol{y}-\boldsymbol{x}\|^2 - \log f(\boldsymbol{x}), \quad (2)\end{aligned}$$

where the first term in (2) is due to the Gaussian distribution of $f(\boldsymbol{y}|\boldsymbol{x})$, and $f(\boldsymbol{x})$ is the prior distribution of the latent clean image.

Finding the prior for the latent clean image is difficult because of its high dimensionality and computational intractability. Patch-based priors alleviate this problem by considering the distribution of the patches instead of the whole image. Letting $\boldsymbol{P}_i \in \mathbb{R}^{d\times n}$ be a patch-extract operator which extracts the ith d-dimensional patch from the image $\boldsymbol{x}$, one can express the negative log of the image prior as a sum of the log patch priors, leading to the expected patch log-likelihood (EPLL) framework [15]:

$$\underset{\boldsymbol{x}}{\operatorname{argmin}}\ \frac{\sigma^{-2}}{2}\|\boldsymbol{y}-\boldsymbol{x}\|^2 - \frac{1}{n}\sum_{i=1}^{n}\log f(\boldsymbol{P}_i\boldsymbol{x}). \quad (3)$$

To define the distribution $f(\boldsymbol{P}_i\boldsymbol{x})$, in this paper we consider the Gaussian mixture model (GMM), although other mixtures of the exponential family are equally applicable. Denoting $\boldsymbol{p}_i \overset{\text{def}}{=} \boldsymbol{P}_i\boldsymbol{x}$ as the ith patch, the distribution $f(\boldsymbol{p}_i)$ is

$$f(\boldsymbol{p}_i) = \sum_{k=1}^{K}\ \pi_k\mathcal{N}(\boldsymbol{p}_i \,|\, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad (4)$$

where π_k, $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ are the weight, mean and covariance of the kth Gaussian component, respectively. For notational simplicity we collectively denote them as $\boldsymbol{\Theta} \overset{\text{def}}{=} \{(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}_{k=1}^K$. Learning the parameter $\boldsymbol{\Theta}$ can be done using the Expectation-Maximization (EM) algorithm. We shall not repeat the EM algorithm here. Interested readers can refer to [23] for a comprehensive tutorial.

2.2. Denoising with Gaussian Mixtures

With the GMM defined in (4), one can solve the optimization in (3) by using the *half quadratic splitting* strategy [24, 25]. The idea is to consider an equivalent problem

$$\underset{\boldsymbol{x},\{\boldsymbol{v}_i\}}{\operatorname{argmin}}\ \frac{n\sigma^{-2}}{2}\|\boldsymbol{y}-\boldsymbol{x}\|^2 + \sum_{i=1}^{n}\left(-\log f(\boldsymbol{v}_i) + \frac{\beta}{2}\|\boldsymbol{P}_i\boldsymbol{x}-\boldsymbol{v}_i\|^2\right), \quad (5)$$

and solve for $\boldsymbol{x}$ and $\{\boldsymbol{v}_i\}_{i=1}^n$ alternatingly. If we further assume that $f(\boldsymbol{v}_i)$ is dominated by one of the components k_i^* where

$$k_i^* \overset{\text{def}}{=} \underset{k}{\operatorname{argmax}}\ \ \pi_k\mathcal{N}(\boldsymbol{v}_i \,|\, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k),$$

then the alternating minimization in (5) can be simplified as

$$\begin{aligned}\boldsymbol{x} &= \left(n\sigma^{-2}\boldsymbol{I} + \beta\sum_{i=1}^{n}\boldsymbol{P}_i^T\boldsymbol{P}_i\right)^{-1}\left(n\sigma^{-2}\boldsymbol{y} + \beta\sum_{i=1}^{n}\boldsymbol{P}_i^T\boldsymbol{v}_i\right), \\ \boldsymbol{v}_i &= \left(\beta\boldsymbol{\Sigma}_{k_i^*} + \boldsymbol{I}\right)^{-1}\left(\boldsymbol{\mu}_{k_i^*} + \beta\boldsymbol{\Sigma}_{k_i^*}\boldsymbol{P}_i\boldsymbol{x}\right). \quad (6)\end{aligned}$$

3. EM ADAPTATION

3.1. Limitations of the EM Algorithm

While the above denoising framework produces reasonable results, learning the Gaussian mixture model has a number of issues.

1. **Adaptivity**: The GMM learned by the standard EM algorithm is a *global* structure of *all* the images in the database. The GMM is thus a *generic* prior as it is not specifically learned for a particular image. Previous works have shown that performance would improve if one uses a targeted database [18, 19]. However, how to adaptively learn a prior from an external database to a specific image remains unclear.
2. **Computational cost**: Learning a good GMM requires a large number of training samples. It is a computationally expensive task, requiring weeks of work on clusters [15].
3. **Finite samples**: When training samples are few, the learned GMM will be over-fitted, and some components will even become singular.
4. **Noise**: Most learning-based methods are designed to learn statistics from *clean* images. However, if we are only given a *noisy* image, it becomes unclear how one can learn the prior effectively.

3.2. The EM Adaptation Algorithm

Our proposed EM adaptation algorithm is a modification of the standard EM algorithm. Suppose that from an external database we have already learned a GMM with parameter $\boldsymbol{\Theta} \overset{\text{def}}{=} \{(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}_{k=1}^K$. The goal now is to learn a specific GMM with parameters $\widetilde{\boldsymbol{\Theta}} \overset{\text{def}}{=} \{(\widetilde{\pi}_k, \widetilde{\boldsymbol{\mu}}_k, \widetilde{\boldsymbol{\Sigma}}_k)\}_{k=1}^K$ so that $\widetilde{\boldsymbol{\Theta}}$ is a good fit to the specific image as well as the external database.

To take a closer look at the EM adaptation algorithm, we denote $\{\widetilde{\boldsymbol{p}}_1, \ldots, \widetilde{\boldsymbol{p}}_n\}$ as the collection of patches taken from the specific image. The E-step and the M-step are modified as follows.

E-Step: In the E-step of the EM adaptation, we compute the likelihood of $\widetilde{\boldsymbol{p}}_i$ conditioned on the generic parameter $\boldsymbol{\Theta}$. This returns a normalized probability

$$\gamma_{ki} = \frac{\pi_k\mathcal{N}(\widetilde{\boldsymbol{p}}_i \,|\, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^{K}\pi_l\mathcal{N}(\widetilde{\boldsymbol{p}}_i \,|\, \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}, \quad (7)$$

and we can define an intermediate parameter $n_k = \sum_{i=1}^n \gamma_{ki}$.

M-Step: The more interesting step of the adaptation is the M-step. Normally, without the EM adaptation, the M-step updates $(\pi_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ using only the available data, which is $\{\widetilde{\boldsymbol{p}}_1, \ldots, \widetilde{\boldsymbol{p}}_n\}$ in our case. However, with the generic prior we modify the M-step by re-weighting the contributions of the generic parameter and the new data. Taking the mean as an example, the EM adaptation updates the mean according to

$$\widetilde{\boldsymbol{\mu}}_k = \underbrace{\alpha_k\left(\frac{1}{n_k}\sum_{i=1}^{n}\gamma_{ki}\widetilde{\boldsymbol{p}}_i\right)}_{\textbf{new data}} + \underbrace{(1-\alpha_k)\boldsymbol{\mu}_k}_{\textbf{generic prior}}, \quad (8)$$

where $\alpha_k \overset{\text{def}}{=} \frac{n_k}{n_k+\rho}$ is the combination weight, with a predefined relevance factor ρ. (8) suggests that the adaptation is in fact a *soft* combination of the generic prior and the new data (as opposed to the

Algorithm 1 EM-adaptation Algorithm

Input: $\boldsymbol{\Theta} = \{(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}_{k=1}^K$, $\{\widetilde{\boldsymbol{p}}_1, \ldots, \widetilde{\boldsymbol{p}}_n\}$.
Output: Adapted parameters $\widetilde{\boldsymbol{\Theta}} = \{(\widetilde{\pi}_k, \widetilde{\boldsymbol{\mu}}_k, \widetilde{\boldsymbol{\Sigma}}_k)\}_{k=1}^K$.
E-step : Compute, for $k = 1, \ldots, K$ and $i = 1, \ldots, n$

$$\gamma_{ki} = \frac{\pi_k \mathcal{N}(\widetilde{\boldsymbol{p}}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^K \pi_l \mathcal{N}(\widetilde{\boldsymbol{p}}_i | \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}, \quad n_k = \sum_{i=1}^n \gamma_{ki}. \tag{9}$$

M-step : Compute, for $k = 1, \ldots, K$

$$\widetilde{\pi}_k = \alpha_k \frac{n_k}{n} + (1 - \alpha_k)\pi_k, \tag{10}$$

$$\widetilde{\boldsymbol{\mu}}_k = \alpha_k \frac{1}{n_k} \sum_{i=1}^n \gamma_{ki} \widetilde{\boldsymbol{p}}_i + (1 - \alpha_k)\boldsymbol{\mu}_k, \tag{11}$$

$$\begin{aligned} \widetilde{\boldsymbol{\Sigma}}_k = \; & \alpha_k \frac{1}{n_k} \sum_{i=1}^n \gamma_{ki} (\widetilde{\boldsymbol{p}}_i - \widetilde{\boldsymbol{\mu}}_k)(\widetilde{\boldsymbol{p}}_i - \widetilde{\boldsymbol{\mu}}_k)^T \\ & + (1 - \alpha_k) \left(\boldsymbol{\Sigma}_k + (\boldsymbol{\mu}_k - \widetilde{\boldsymbol{\mu}}_k)(\boldsymbol{\mu}_k - \widetilde{\boldsymbol{\mu}}_k)^T \right). \end{aligned} \tag{12}$$

Postprocessing: Normalize $\{\widetilde{\pi}_k\}_{k=1}^K$ so that they sum to 1, and ensure $\{\widetilde{\boldsymbol{\Sigma}}_k\}_{k=1}^K$ is positive semi-definite.

hard combination in the fusion methods). If $\alpha_k \to 1$, then more emphasis will be put on the new data. Conversely, if $\alpha_k \to 0$, then we will keep the generic parameter. The swing between the two options is automatically controlled by the likelihood sum $n_k = \sum_{i=1}^n \gamma_{ki}$.

Other steps (i.e., learning $\widetilde{\pi}_i$ and $\widetilde{\boldsymbol{\Sigma}}_i$) of the EM adaptation are summarized in Algorithm 1. We remark that the EM adaptation consists of only *one* step of the standard EM algorithm. Additional iterations can still be implemented. However, in practice we find that most adaptation converges in one iteration.

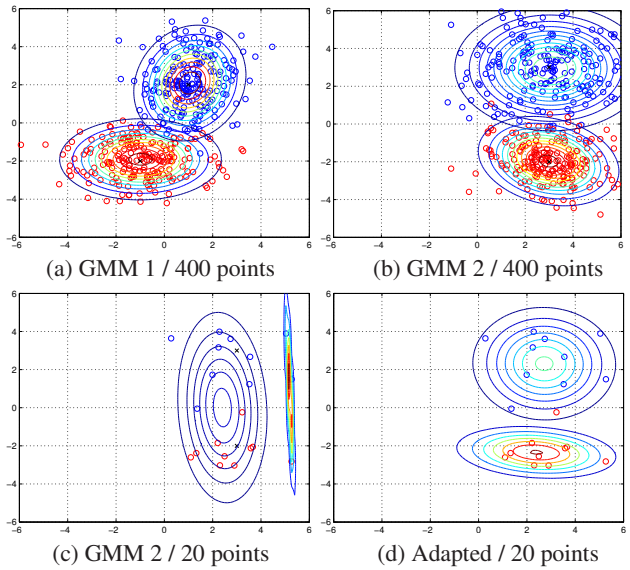
Toy Example: We illustrate the idea of the EM adaptation using a toy example. Figure 1 (a) and (b) show two individual GMMs learned from two sets of data, each containing 400 data points. It can be seen that the GMMs are estimated reasonably well. In (c), we draw a subset of 20 data points from (b) and try to learn a GMM from these 20 data points. Immediately we observe that the learned GMM is over-fitted to these 20 data points, and is very different from the GMM in (b). Using the EM adaptation, we observe in (d) that the adapted GMM is significantly better than (c), despite the fact that it only uses 20 data points.

Remark: The EM adaptation algorithm presented above can be theoretically derived from a Bayesian perspective for a hidden Markov model. The weights $\widetilde{\pi}_k$ are the parameters of a Dirichlet distribution, and $(\widetilde{\boldsymbol{\mu}}_k, \widetilde{\boldsymbol{\Sigma}}_k)$ are the parameters of the normal-Wishart distribution. For details we refer the readers to [26]. In literature, the EM-adaptation has also been applied to speaker identification [27, 28] and image classification [29].

3.3. Adaptation to a Pre-filtered Image

At this point, careful readers would probably argue that Algorithm 1 will only work for *clean* images. We now discuss how the algorithm can be modified to learn a GMM using *noisy* images.

In the presence of noise, we adopt a pre-filtering approach similar to the two-stage denoising methods such as BM3D and EPLL. In the first stage, we apply an existing denoising algorithm to obtain a pre-filtered image. The adaptation is then applied in the second stage to the pre-filtered image. However, since a pre-filtered image is not the same as the latent clean image, we must quantify the residual noise remaining in the pre-filtered image.



(a) GMM 1 / 400 points (b) GMM 2 / 400 points (c) GMM 2 / 20 points (d) Adapted / 20 points

Fig. 1. (a) and (b): Two GMMs, each learned from 400 data points. (c): A GMM learned from a subset of 20 data points drawn from (b). (d): An adapted GMM using the same 20 data points in (c). Note the significant improvement from (c) to (d) by using the proposed adaptation. In this example we let $\rho = 1$.

To this end, we let $\bar{\boldsymbol{x}}$ be the pre-filtered image. Our objective is to estimate the mean squared error remaining in $\bar{\boldsymbol{x}}$, i.e., $\mathbb{E}\|\bar{\boldsymbol{x}} - \boldsymbol{x}\|^2$. In this paper we use the Stein's Unbiased Risk Estimator [30, 31]. Due to space limit we shall leave the details to a follow up article. Once the mean squared error is estimated, we model the residual noise as a single Gaussian so that $(\bar{\boldsymbol{x}} - \boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{0}, \widetilde{\sigma}^2 \boldsymbol{I})$, where $\widetilde{\sigma}^2$ is the estimated mean squared error. Consequently, we modify (9) as

$$\gamma_{ki} = \frac{\pi_k \mathcal{N}(\widetilde{\boldsymbol{p}}_i \,|\, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k + \widetilde{\sigma}^2 \boldsymbol{I})}{\sum_{l=1}^K \pi_l \mathcal{N}(\widetilde{\boldsymbol{p}}_l \,|\, \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l + \widetilde{\sigma}^2 \boldsymbol{I})}, \tag{13}$$

and (12) as

$$\begin{aligned} \widetilde{\boldsymbol{\Sigma}}_k = \; & \alpha_k \frac{1}{n_k} \sum_{i=1}^n \gamma_{ki} \left((\widetilde{\boldsymbol{p}}_i - \widetilde{\boldsymbol{\mu}}_k)(\widetilde{\boldsymbol{p}}_i - \widetilde{\boldsymbol{\mu}}_k)^T - \widetilde{\sigma}^2 \boldsymbol{I} \right) \\ & + (1 - \alpha_k) \left(\boldsymbol{\Sigma}_k + (\boldsymbol{\mu}_k - \widetilde{\boldsymbol{\mu}}_k)(\boldsymbol{\mu}_k - \widetilde{\boldsymbol{\mu}}_k)^T \right), \end{aligned} \tag{14}$$

which perturbs the covariance by the noise remaining in the pre-filtered image $\bar{\boldsymbol{x}}$.

4. EXPERIMENTAL RESULTS

4.1. Experiment Settings

We compare our proposed method with BM3D [3] and EPLL [15], which are among the best performing methods for image denoising. The default GMM in EPLL is learned from 2,000,000 randomly chosen 8×8 patches. This default GMM is used as the generic prior for the proposed EM adaptation.

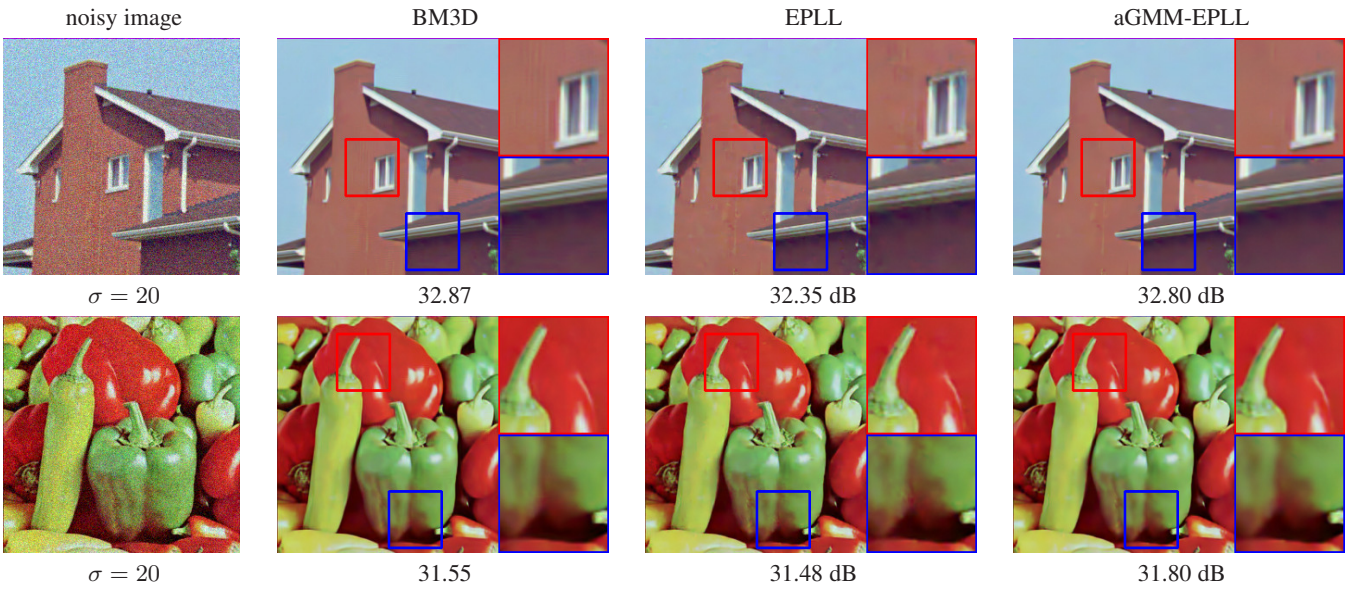


Fig. 2. Image denoising with the adapted GMM on the EPLL denoised image: Visual comparison and objective comparison. The test images are of size 256×256. Denoising is performed for each color channel individually.

		BM3D	aGMM -BM3D	EPLL	aGMM -EPLL	aGMM -clean
Boat	$\sigma = 20$	29.59	29.87	29.78	**29.94**	30.47
	$\sigma = 40$	26.19	26.66	26.54	**26.70**	27.01
	$\sigma = 60$	24.65	24.72	24.74	**24.85**	25.11
	$\sigma = 80$	23.39	23.27	23.38	**23.40**	23.62
	$\sigma = 100$	22.60	22.41	22.60	**22.58**	22.77
Cameraman	$\sigma = 20$	30.24	30.34	30.24	**30.40**	31.33
	$\sigma = 40$	26.83	**27.35**	26.98	27.31	27.97
	$\sigma = 60$	25.31	25.36	25.30	**25.54**	26.25
	$\sigma = 80$	**23.88**	23.77	23.70	**23.88**	24.53
	$\sigma = 100$	**22.95**	22.83	22.75	22.90	23.48
House	$\sigma = 20$	33.65	**33.77**	32.98	33.56	34.58
	$\sigma = 40$	30.47	**30.84**	29.90	30.71	31.54
	$\sigma = 60$	**28.60**	28.36	27.64	28.32	29.02
	$\sigma = 80$	**27.36**	27.18	26.61	27.18	27.75
	$\sigma = 100$	**25.75**	25.68	25.18	25.64	25.96
Lena	$\sigma = 20$	31.61	31.76	31.44	**31.79**	32.68
	$\sigma = 40$	27.83	28.23	28.07	**28.36**	28.81
	$\sigma = 60$	**26.38**	26.22	26.04	26.31	26.59
	$\sigma = 80$	**24.93**	24.83	24.53	24.83	25.04
	$\sigma = 100$	**23.99**	23.89	23.63	23.91	24.03
Montage	$\sigma = 20$	**33.57**	33.52	32.55	33.23	34.63
	$\sigma = 40$	29.13	**29.39**	28.37	29.06	30.21
	$\sigma = 60$	26.56	**26.66**	25.97	26.59	27.60
	$\sigma = 80$	24.96	**24.98**	24.38	24.95	25.74
	$\sigma = 100$	23.72	**23.76**	23.27	23.64	24.48
Peppers	$\sigma = 20$	31.23	31.50	31.23	**31.52**	32.40
	$\sigma = 40$	27.43	27.96	27.72	**28.03**	28.52
	$\sigma = 60$	25.78	25.87	25.68	**25.99**	26.38
	$\sigma = 80$	24.27	24.45	24.11	**24.50**	24.70
	$\sigma = 100$	23.05	23.28	22.91	**23.30**	23.58
Average		26.86	**26.96**	26.61	**26.96**	27.56

Table 1. PSNR results for standard images of size 256×256

We consider three versions of EM-adaptation: (1) An oracle adaptation by adapting the generic prior to the ground truth clean image, denoted as *aGMM-clean*; (2) A pre-filtered adaptation by adapting the generic prior to the EPLL result, denoted as *aGMM-EPLL*; (3) A pre-filtered adaptation by adapting the generic prior to the BM3D result, denoted as *aGMM-BM3D*.

4.2. Denoising Standard Test Images

We studied 6 images of size 256×256 in our experiments. Figure 2 shows the results of two testing images. It can be seen that the proposed method yields the best result in terms of PSNR values. Visually, our method preserves the texture details. In Table 1, we report the detailed PSNR values for different noise variances. Comparing aGMM-EPLL with EPLL, we observe that the proposed aGMM-EPLL is consistently better than EPLL with an average gain of more than 0.3 dB. In some cases, aGMM-EPLL is even better than BM3D, e.g., Boat, Lena and Peppers. If we further look at the oracle aGMM-clean, the performance gain is even higher. We remark that all the adapted GMMs are learned from the pre-filtered images. If one uses the standard EM algorithm to learn a GMM from the pre-filtered image, the result will be several dBs worse than the adapted ones.

5. CONCLUSION

We proposed an EM adaptation method to learn image priors. The proposed adaptation method allows us to use the parameters of a generic Gaussian mixture model and adapt to the image of interest. Compared to the classic EM algorithm, the proposed EM adaptation requires significantly less training samples and could be applied to any pre-filtered image. We also proposed modifications to compensate for the noise remaining in the pre-filtered images. EM adaptation is a computationally efficient method. Experimental results showed that the proposed method outperforms EPLL and BM3D. Detailed analysis of the algorithm and automated estimation of the internal parameters will be studied in our future work.

6. REFERENCES

[1] A. Buades, B. Coll, and J. Morel, "A review of image denoising algorithms, with a new one," *SIAM Multiscale Model and Simulation*, vol. 4, no. 2, pp. 490–530, 2005.

[2] C. Kervrann and J. Boulanger, "Local adaptivity to variable smoothness for exemplar-based image regularization and representation," *International Journal of Computer Vision*, vol. 79, no. 1, pp. 45–69, 2008.

[3] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "A nonlocal and shape-adaptive transform-domain collaborative filtering," in *Proc. Intl. Workshop on Local and Non-Local Approx. in Image Process. (LNLA'08)*, Aug. 2008, pp. 1–8.

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "BM3D image denoising with shape-adaptive principal component analysis," in *Signal Process. with Adaptive Sparse Structured Representations (SPARS'09)*, Apr. 2009, pp. 1–6.

[6] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR'09)*, Sep. 2009, pp. 2272–2279.

[7] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principal component analysis with local pixel grouping," *Pattern Recognition*, vol. 43, pp. 1531–1549, Apr. 2010.

[8] W. Dong, L. Zhang, G. Shi, and X. Li, "Nonlocally centralized sparse representation for image restoration," *IEEE Trans. Image Process.*, vol. 22, no. 4, pp. 1620 – 1630, Apr. 2013.

[9] A. Rajwade, A. Rangarajan, and A. Banerjee, "Image denoising using the higher order singular value decomposition," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 35, no. 4, pp. 849 – 862, Apr. 2013.

[10] S. H. Chan, T. Zickler, and Y. M. Lu, "Monte Carlo nonlocal means: Random sampling for large-scale image filtering," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3711–3725, Aug. 2014.

[11] M. Zontak and M. Irani, "Internal statistics of a single natural image," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR'11)*, Jun. 2011, pp. 977–984.

[12] I. Mosseri, M. Zontak, and M. Irani, "Combining the power of internal and external denoising," in *Proc. Intl. Conf. Computational Photography (ICCP'13)*, Apr. 2013, pp. 1–9.

[13] H. C. Burger, C. J. Schuler, and S. Harmeling, "Learning how to combine internal and external denoising methods," *Pattern Recognition*, pp. 121–130, 2013.

[14] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[15] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *Proc. IEEE Intl. Conf. Computer Vision (ICCV'11)*, Nov. 2011, pp. 479–486.

[16] A. H. Sayed, *Fundamentals of Adaptive Filtering*, New York: Wiley, 2003.

[17] E. Luo, S. H. Chan, S. Pan, and T. Q. Nguyen, "Adaptive nonlocal means for multiview image denoising: Searching for the right patches via a statistical approach," in *Proc. IEEE Intl. Conf. Image Process. (ICIP'13)*, Sep. 2013, pp. 543–547.

[18] E. Luo, S. H. Chan, and T. Q. Nguyen, "Image denoising by targeted external databases," in *Proc. IEEE Intl. Conf. Acoustics, Speech and Signal Process. (ICASSP '14)*, May 2014, pp. 2469–2473.

[19] E. Luo, S. H. Chan, and T. Q. Nguyen, "Adaptive image denoising by targeted databases," *IEEE Trans. Image Process.*, vol. 24, no. 7, pp. 2167–2181, Jul. 2015.

[20] H. Yue, X. Sun, J. Yang, and F. Wu, "CID: Combined image denoising in spatial and frequency domains using web images," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'14)*. IEEE, 2014, pp. 2933–2940.

[21] J. Sulam and M. Elad, "Expected patch log likelihood with a sparse prior," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*. Springer, 2015, pp. 99–111.

[22] G. Yu, G. Sapiro, and S. Mallat, "Solving inverse problems with piecewise linear estimators: From Gaussian mixture models to structured sparsity," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2481–2499, May 2012.

[23] M. R. Gupta and Y. Chen, *Theory and use of the EM algorithm*, Now Publishers Inc, 2011.

[24] D. Geman and C. Yang, "Nonlinear image recovery with half-quadratic regularization," *IEEE Trans. Image Process.*, vol. 4, no. 7, pp. 932–946, Jul. 1995.

[25] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-Laplacian priors," in *Advances in Neural Information Processing Systems 22*, pp. 1033–1041. Curran Associates, Inc., 2009.

[26] J. Gauvain and C. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions Speech and Audio Process.*, vol. 2, no. 2, pp. 291–298, Apr. 1994.

[27] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital signal process.*, vol. 10, no. 1, pp. 19–41, 2000.

[28] P. C. Woodland, "Speaker adaptation for continuous density hmms: A review," in *In ITRW on Adaptation Methods for Speech Recognition*, Aug. 2001, pp. 11–19.

[29] M. Dixit, N. Rasiwasia, and N. Vasconcelos, "Adapted gaussian models for image classification," in *IEEE Conference Computer Vision and Pattern Recognition (CVPR'11)*, Jun. 2011, pp. 937–943.

[30] C. M. Stein, "Estimation of the mean of a multivariate normal distribution," *The Annals of Statistics*, vol. 9, pp. 1135–1151, 1981.

[31] S. Ramani, T. Blu, and M. Unser, "Monte-Carlo SURE: A black-box optimization of regularization parameters for general denoising algorithms," *IEEE Trans. on Image Process.*, vol. 17, no. 9, pp. 1540–1554, Sep. 2008.