A Patch Aware Multiple Dictionary Framework for Demosaicing

Meiqing Zhang, Linmi Tao

Department of Computer Science and Technology, Tsinghua University

Abstract. Most digital cameras rely on demosaicing algorithms to restore true color images. Data captured by these cameras is reduced by two thirds because a Color Filter Array (CFA) allows only one particular channel go through at each pixel. This paper proposes a Patch Aware Multiple Dictionary (PAMD) framework for demosaicing. Instead of using a common dictionary, multiple dictionaries are trained for different classes of signals. These class-specific dictionaries comprise a super overcomplete dictionary. The most suitable dictionary would be adaptively selected based on the patch class, determined by the *Energy Exclusiveness Feature* (EEF) which measures the degree of energy domination in the representation code. In this way, candidate atoms are constrained in a set of atoms with low correlations; and meanwhile, the signal would have sparser representation over this adapted dictionary than over the common one, making the fixed sample rate relatively high and thus, accomplishing satisfying restoration according to the compressive sensing theory. Extensive experiments demonstrated that PAMD outperforms traditional Single Dictionary (SD) based approach as well as leading algorithms in diffusion-based family significantly, with respect to both PSNR and visual quality. Especially, the general artifact by Bayer CFA, Moiré Pattern, is dramatically reduced. Furthermore, on several images, it also significantly outperforms the state-of-the-art algorithms which are also sparse coding based but very complicated. PAMD is a general framework which can cooperate with existing demosaicing algorithms based on sparse coding.

1 Introduction

Demosaicing is a key technique in most digital cameras to recover true color images from data mosaiced by Color Filter Array (CFA) which allows only one channel go through at each pixel. Its great impact on numerous consumer cameras (especially on medium- and low-end devices such as mobiles) has drawn great attentions to demosaicing research in the past decade [1,2,3,4,5,6]. Numerous algorithms have been proposed for demosaicing. They can be roughly categorized as classical diffusion-based [7,8,9] or sparse coding based [1,10,11,12]. Diffusion-based approaches perform filling-in at pixel level by diffusing information from known regions to missing parts, typical examples include the widely used interpolation family algorithms such as OAP [4], refined ACPI [13] and kernel regression based interpolation [14].

Sparse coding based approaches treat the demosaicing problem from a different perspective. Prior knowledge is packed in dictionaries and under the assumption that signals of interests can be represented using a small number of atoms (i.e., sparsely representable). Compressive sensing theory [15], on the other hand, provides solid theory foundations for accurate recovery of sparse signal from only a few observations [1,16]. Generally, the dictionary can be comprised with mathematically pre-defined bases (e.g., [1]) or learnt from training samples (e.g., [10,11]). One of the advantages of educated dictionary is that its knowledge is from domain-specific training samples thus can be particularly efficient [17]. This kind of approaches has achieved stat-of-the-art demosaicing results [1,10].

The success of sparse coding based approaches evidences that sparsity should be learned for specific signal. Guided by this philosophy, this paper proposes the Patch Aware Multiple Dictionary (PAMD) framework for demosaicing. In learning stage, patches are classified and a dictionary is trained for each class; then in demosaicing stage, the most suitable dictionary would be automatically selected according to the patch class. In this way, it makes the selected dictionary adaptive to the signal characteristic and improves the sparsity level of the representation code, as evidenced by our experiments. Consequently, better demosaicing performance is achieved.

PAMD allows multiple dictionaries, rather than one, to be trained from numerous samples, thus can pack more knowledge. These dictionaries comprise a super overcomplete dictionary; and meanwhile, candidate atoms for sparse coding are still constrained in a set where low correlation between atoms is ensured, which is a basic premise of most sparse coding algorithms. Our extensive experiments demonstrated that PAMD outperforms traditional Single Dictionary (SD) based approach as well as leading algorithms in diffusion-based family significantly, with respect to both PSNR and visual quality. Furthermore, on several images, it also significantly outperforms the state-of-the-art algorithms which are also sparse coding based but very complicated. Especially, the general artifact by Bayer CFA, Moiré Pattern, is dramatically reduced. PAMD is a general framework which can cooperate with current sparse coding based state-of-the-art algorithms.

2 Problem Formulation, Related Work and Contributions

2.1 Problem formulation

Suppose that the initial color image $\boldsymbol{X}$ is mosaiced to be $\boldsymbol{Y}$, as formulated in Equation (1). $\boldsymbol{M}$ is a known matrix determined by the spatial layout of missing pixels (e.g., Bayer pattern).

$$\boldsymbol{Y} = \boldsymbol{M}\boldsymbol{X} \tag{1}$$

Demosaicing aims to estimate $\hat{\boldsymbol{X}}$ from $\boldsymbol{Y}$. It is usually performed in vectorized patch space. Patch vector $\hat{\boldsymbol{x}} \in R^{3n}$ is estimated based on the observation $\boldsymbol{y} \in R^n$.

The core assumption used in estimating $\boldsymbol{x} \in R^{3n}$ is that it has a sparse representation over a dictionary $\boldsymbol{D} \in R^{3n \times L}$ ($L \geq 3n$), a sparse coding problem as formulated in Equation (2) where $||\boldsymbol{s}||_0$ is the ℓ^0 norm (i.e, the sparsity).

$$\boldsymbol{x} \approx \boldsymbol{D}\boldsymbol{s} \; subject\ to \; ||\boldsymbol{s}||_0 \ll L \tag{2}$$

Suppose that $\boldsymbol{y}$ can be written as $\boldsymbol{y} = \boldsymbol{M}_t\boldsymbol{x}$, then combining Equation (2) we have Equation (3).

$$\boldsymbol{y} = \boldsymbol{M}_t\boldsymbol{x} \approx \boldsymbol{M}_t\boldsymbol{D}\boldsymbol{s}, \;\; subject\ to \; ||\boldsymbol{s}||_0 \ll L \tag{3}$$

Sparse representation code $\hat{\boldsymbol{s}}$ of $\boldsymbol{y}$ over the dictionary $\boldsymbol{M}_t\boldsymbol{D}$ can be first computed as shown in Equation (4). Then $\hat{\boldsymbol{x}}$ can be estimated as $\boldsymbol{D}\hat{\boldsymbol{s}}$.

$$\hat{\boldsymbol{s}} = \arg\min ||\boldsymbol{s}||_p \; subject\ to \; ||\boldsymbol{y} - (\boldsymbol{M}_t\boldsymbol{D})\boldsymbol{s}||_2^2 < \xi \tag{4}$$

Equation (4) can be written in the regularized form as Equation (5).

$$\hat{\boldsymbol{s}} = \arg\min ||\boldsymbol{y} - (\boldsymbol{M}_t\boldsymbol{D})\boldsymbol{s}||_2^2 + \lambda||\boldsymbol{s}||_p \tag{5}$$

Ideally, p in Equation (4) should be taken as 0, the same as that in Equation (3). However, directly minimizing the ℓ^0 norm is NP-hard. Alternatives include the Smoothed L0 algorithm (SL0) [18], which tries to directly minimize the ℓ^0 norm by using a continuous smooth function to approximate the discrete ℓ^0 norm; and greedy methods such as Orthogonal Matching Pursuit (OMP) [19]. Other classical alternative solutions are ℓ^1 norm based approaches such as L1 magic [20], feature-sign search [21] and LASSO [22].

Though efficient sparse coding algorithms are critical, the most fundamental ingredient is the basis set, i.e., the dictionary. For example, if the cosine distances between the signal and all of the bases are large, it would make most coefficients large and sparse representation is hardly possible.

Consequently, it is better to learn the dictionary $\boldsymbol{D}$ from samples than to use pre-defined mathematical bases [11,23], so that the signals of the specific class are sparse in the space spanned by these bases.

Generally, ℓ^0 regularization based applications (e.g., sparse classification) is more eager for a data-class-specific and educated dictionary than ℓ^1 and ℓ^2, because the number of its activated atoms is limited, the fitting error would be large if there are no bases which are cosine similar to the signal. Popular algorithms for obtaining an educated dictionary from training sample matrix include K-SVD [11], NMF [24] and Independent Component Analysis (ICA) [25,26].

Besides learning from samples, another method to improve the sparsity level is increasing the number of atoms, constructing an overcomplete dictionary ($L > 3n$). However, the fatal counterproductive effect is the increasing of correlations among atoms, which breaks the premise of most sparse coding algorithms.

PAMD tackles this problem by patch classification and training a dictionary for each class. Candidate atoms for sparse coding are still constrained in only one of the dictionaries thus low correlations are ensured. Meanwhile, in class-specific dictionary, it is more possible to fit the signal using only a small number of atoms, thus improving the sparsity level of the representation code.

2.2 Relation with Previous Work and Contributions

The leading demosaicing algorithms in diffusion based family include OAP [4], DL [12], LPA [27], CAD [3] and regularization based approach [28]. The *a priori* about natural images used in these approaches are usually based on pre-defined rules, rather than data driven. For example, the assumptions of channel smoothness [28] and local similarity [27]. A comprehensive survey can be found in [29].

Sparse coding and Compressive Sensing (CS) theory [30,15] provide a different perspective for demosaicing [1,10]. Sparse coding based approaches exploit the *a priori* that natural images are sparse after suitable transformation. Standard CS theory has shown that robust reconstruction of full signal $\boldsymbol{s}$ (or $\boldsymbol{x}$) from $\boldsymbol{y}$ which contains only a few observations is possible, as long as the transformation and observation matrices satisfy certain conditions and the number of measurements m meets the criterion that

$$m > c||\boldsymbol{s}||_0 log(L/||\boldsymbol{s}||_0), \tag{6}$$

where c is a small constant [16]. If the signal is not sparse enough in the transformed space, or the sample rate is too low, the sparse coding based demosaicing approaches would not yield satisfying results [1].

The sparsity assumption is found to be very efficient for demosaicing, and has achieved state-of-the-art results [1,10]. According to Equation (6), because educated dictionary from samples can provide sparser representation than predefined mathematical bases, the fixed sample rate $1/3$ would become relatively high, so satisfying restoration performance can be achieved.

The successes of current state-of-the-art algorithms [1,10] rely on various techniques. In [10], an online learning algorithm [31] is exploited to train a highly optimized dictionary from numerous samples (more than 2×10^7 patches); then another dictionary which packs the non-local similarity and group sparsity constraints is trained on-the-fly based on patches in the test image which are similar to the patch to be demosaiced. These two dictionaries are concatenated for restoring the patch. This approach provides a more overcomplete dictionary and meanwhile, imposes the non-local similarity constraints on the sparse coding, thus it would be very efficient for images which contain many similar patches. However, it will cause heavy computation overhead in testing stage.

In [1], adaptive patch sizes are used, rather than fixed. It is observed that small patch size is more suitable for singular area while large size for texture and smooth areas [1]. A high-pass filter is trained to filter each large patch, to detect whether it should be divided into small patches based on the energy of the filtered patch.

PAMD is much simpler than them and has low computation overhead in testing stage. No training is needed in testing. Based on the simple *Energy Exclusiveness Feature* (EEF) and fast classification method, it trains a dictionary for each class using samples falling in the same class, and the mosaiced patch is restored using the dictionary corresponding to its class.

To reveal the relationship between our approach and previous work, we interpret [10,1] also from the perspective of PAMD. In [10,1], the concept of patch

class only exists in testing stage. In [10], patches in the test image are clustered into many classes based on intensities, and a dictionary is trained using patches in each class. Multiple dictionary training while testing involves additional high computation load. Moreover, the amount of patches used in LSSC is very large, more than 2×10^7, significantly larger than the average number of about 4×10^4 for each dictionary in PAMD. In [1], adaptive patch size is initially introduced to reduce the artifacts. From the view of PAMD, in the test image, high-pass filtered patches having the energy above the threshold would be in one class while others below the threshold are in the other class.

To sum up, the main contributions of this paper can be outlined as below.

a) The PAMD framework is proposed to furthermore refine the patch class in both training and testing stages, so that the representation coefficient vector would become sparser, making the fixed sample rate relatively higher and then yielding more satisfying restoration.
b) The *Energy Exclusiveness Feature* (EEF) is proposed for patch classification. EEF is a scalar so simple that classification just involves bin location. PAMD has low computation overhead in testing.
c) Extensive experiments showed that PAMD significantly improves the demosaicing performance over the single dictionary based approach on standard benchmark, with respect to both PSNR and visual quality. Especially, the general artifact by Bayer CFA, Moiré Pattern, is dramatically reduced. It also outperforms the leading algorithms in diffusion based algorithm family. Furthermore, though PAMD is very simple, its performance is comparable with the complicated state-of-the-art algorithms based on sparse coding, and even better than them on several images.

Moreover, PAMD can corporate with existing sparse coding based algorithms. Based on patch classification, they can be performed in each class. Sparser representation is a common advantage, PAMD is a general framework and can be potentially utilized in other fields of applications.

3 Patch Aware Multiple Dictionary Framework

3.1 Energy Exclusiveness Feature and Patch Classification

Energy Exclusiveness Feature (EEF) is a scalar f which measures the magnitude of a patch follows the power law phenomenon after the sparse transformation, i.e., a small number of sparse representation coefficients dominate most of the energy. Suppose that the coefficient vector of the patch over the dictionary is $\boldsymbol{s} = (s_1, \cdots, s_L)^T$, and $(d_1, \cdots, d_L)$ is the index permutation so that $|s_{d_i}|^p$ $(p > 0)$ is monotone non-increasing, and $\tau \in (0, 1)$ is a threshold, then f is defined as Equation (7).

$$f \triangleq k/L \ , k = \arg\min \frac{\Sigma_{i=1}^{k} |s|_{d_i}^p}{\Sigma_{i=1}^{L} |s|_{d_i}^p} \geq \tau \tag{7}$$

The parameter p exponentially magnifies (when $p > 1$) or shrinks ($p < 1$) the difference between coefficients. In our experiments, p is set as 2.0 and τ is taken as 0.7. It can be seen that $f \in [\frac{1}{L}, 1.0]$. Based on EEF, patch classification is very simple. By defining threshold intervals $[\zeta_0, \zeta_1), [\zeta_1, \zeta_2), \cdots, [\zeta_n, 1]$, the class of a patch is determined by the interval in which its EEF falls.

EEF measures the sparsity degree of a patch over the common dictionary. Higher sparsity (smaller f) is important for good reconstruction performance because it means lower minimum sample rate, then the fixed sample rate 1/3 in the demosaicing application would become relatively higher. Our experiments found that a common dictionary is not optimal to provide sparse representation for various patches. Separately training a dictionary for each patch class using samples from the same class would improve the dictionary expression capability for patches in this class, yielding small f.

3.2 EEF based Dictionary Learning and Demosaicing

In the learning stage, vectorized patch samples are concatenated, yielding a whole sample matrix which will be factorized to obtain the common dictionary $\boldsymbol{D}_{all}$. All of the patch samples will be classified into one of r classes based on the EEF of the mosaiced patch over the $\boldsymbol{M}_{rggb}\boldsymbol{D}_{all}$. Here $\boldsymbol{M}_{rggb}$ means "RGGB" mosaic pattern for a patch. Samples in each class j will be used to train a corresponding dictionary $\boldsymbol{D}_j$. Then $[\boldsymbol{D}_1, \cdots, \boldsymbol{D}_r]$ comprise a super overcomplete dictionary. In demosaicing stage, patch class J is determined by the EEF of the mosaiced patch $\boldsymbol{y}_t$ over $\boldsymbol{M}_t\boldsymbol{D}_{all}$. Then dictionary $\boldsymbol{D}_J$ is selected to restore $\boldsymbol{y}_t$, which yields the final demosaiced patch $\hat{\boldsymbol{x}}_t$. Patches are aggregated to a true color image by averaging overlapped pixels. Though atom correlations across $[\boldsymbol{D}_1, \cdots, \boldsymbol{D}_r]$ may be high, PAMD will *adapt* to the patch automatically by constraining candidate atoms within a low-correlated dictionary $\boldsymbol{D}_J$.

The detailed demosaicing algorithm is interpreted as Algorithm 1. Because in training stage, all patches are mosaiced using $\boldsymbol{M}_{rggb}$, however, at each step t in the demosaicing procedure, $\boldsymbol{M}_t$ for the corresponding patch may be not "RGGB" pattern [1]. Our solution is first restoring the whole image using $\boldsymbol{D}_{all}$; secondly, the patch classification is based on the EEF of $\boldsymbol{M}_{rggb}\boldsymbol{x}$ over the dictionary $\boldsymbol{M}_{rggb}\boldsymbol{D}_{all}$, then we compute the code $\hat{\boldsymbol{s}}$ of $\boldsymbol{M}_t\boldsymbol{x}$ over dictionary $\boldsymbol{M}_t\boldsymbol{D}_J$; finally, the patch is estimated as $\hat{\boldsymbol{x}}_t = \boldsymbol{D}_J\hat{\boldsymbol{s}}_t^1$.

Traditionally, averaging weights for a pixel in several sliding windows are the same as $w_t = 1$. However, we propose that the weights can be associated with the EEF f_t as Equation (8).

$$w_t = (1/f_t)^\beta \tag{8}$$

[1] In the setting of Bayer "RGGB" CFA, if the sliding-window moves forward by odd number pixel(s) at each step, then there would be 4 patterns for $\boldsymbol{M}_t$ as "RGGB", "GRBG", "GBRG", and "BGGR". However, if by even number pixels per step, there would be only one pattern as "RGGB", then in this case, we do not have to first restore the whole image using $\boldsymbol{D}_{all}$ in Algorithm 1.

It means that larger weights are given to restored pixels within windows in which patches can be more sparsely represented. The benefits of adaptive weighting policy would be presented and discussed in the experiment section 4.3.

Algorithm 1 PAMD based Demosaicing

Input: $\boldsymbol{D}_{all}$, $\boldsymbol{D}_j$ $(j = 1, \cdots, r)$, mosaiced image $\boldsymbol{Y}$, mosaic matrix $\boldsymbol{M}$ (determined by Bayer CFA), EEF threshold intervals $[\zeta_{j-1}, \zeta_j)$
Output: Demosaiced image $\hat{\boldsymbol{X}}$
1: Pre-restoration: Restore the whole image first using $\boldsymbol{D}_{all}$
▷ `Patch classification is based on the EEF of` $\boldsymbol{M}_{rggb}\boldsymbol{x}$`, so this pre-restoration is necessary ONLY when the sliding window moves forward by odd number pixel(s) per step.`
2: **for** each patch $\boldsymbol{y}_t$ in $\boldsymbol{Y}$ **do**
3: Compute corresponding EEF over $\boldsymbol{M}_{rggb}\boldsymbol{D}_{all}$ and classify this patch as class J
4: Compute $\hat{\boldsymbol{s}}_t$ of $\boldsymbol{y}_t$ over $\boldsymbol{M}_t\boldsymbol{D}_J$ by solving Equation (4)
5: Estimate the patch as $\hat{\boldsymbol{x}}_t = \boldsymbol{D}_J\hat{\boldsymbol{s}}_t$
6: **end for**
7: Aggregate patches $\hat{\boldsymbol{x}}_t$ to obtain $\hat{\boldsymbol{X}}$
▷ `Overlapped pixels would be averaged with weights` w_t`.`

4 Experimental Results

4.1 Settings

Bayer CFA of "RGGB" pattern is used in our experiments, which determines $\boldsymbol{M}_{rggb}$, $\boldsymbol{M}$ in Equation (1) and $\boldsymbol{M}_t$ in Equation (3). Specially, suppose that the patch is of size $2 \times 2 \times 3$ and vectorized to $\boldsymbol{x} \in R^{12}$ in the order as $\boldsymbol{x} = (R_1, R_2, R_3, R_4, G_1, G_2, G_3, G_4, B_1, B_2, B_3, B_4)^T$. Then the mask $\boldsymbol{M}_{rggb}$ is as Equation (9) and $\boldsymbol{y} = (R_1, G_2, G_3, B_4)^T$.

$$\boldsymbol{M}_t = \begin{pmatrix} 1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1 \end{pmatrix} \tag{9}$$

In practice, $\boldsymbol{Y}$ and $\boldsymbol{M}_t\boldsymbol{D}$ are obtained by removing corresponding channels from $\boldsymbol{X}$ and rows from $\boldsymbol{D}$ respectively, instead of constructing the huge $\boldsymbol{M}$.

Training patches of size $p \times q \times 3 = 10 \times 10 \times 3$ are randomly sampled from 122 natural images mainly from McGill Colour Image Database [32]. The average number of training patches for each class is about 4×10^4, a rather small number compared with that of more than 2×10^7 in complicated LSSC [10] whose dictionary training in testing stages causes additional computation overhead.

A common dictionary $\boldsymbol{D}_{all} \in R^{300\times300}$ is factorized using FastICA algorithm [33], an efficient ICA implementation. Then the *energy exclusiveness feature* f is computed from the mosaiced patch over the dictionary $\boldsymbol{M}_{rggb}\boldsymbol{D}_{all} \in$

$R^{100\times300}$. Patches are classified into 4 classes, corresponding f would fall into one of the four bins as $[0, 0.25)$, $[0.25, 0.45)$, $[0.45, 0.65)$ and $[0.65, +\infty)$. So, 4 dictionaries $\boldsymbol{D}_j \in R^{300\times300}(j = 1, \cdots, 4)$ are learned, one for a class.

SL0 [18] is used as the sparse coding algorithm and the decrease factor σ is empirically chosen as 0.4 while the σ_{min} which determines the coding accuracy is set as 0.3. As for PAMD with adaptive averaging weight in Equation (8), parameter β is set as 2.0. Similar with most existing works and for fair comparisons, a boarder of 15 pixels is excluded when calculating PSNR (dB). Test images are from Kodak PhotoCD benchmark [34] which includes 24 color images of 768×512 size. Source code (MatLab implementation) can be found at `http://media.cs.tsinghua.edu.cn/~cvg/zhangmeiqing/PAMD.html`.

4.2 Numerical Results and Visual Quality

The numerical comparisons on Kodak PhotoCD benchmark with respect to PSNR (dB) between our PAMD, Single Dictionary based approach (SD), CAD [3], DL [12] and LPA [27] are presented in Table 1. The images on which PAMD outperforms CD [1] and LSSC [10] can be found in Table 2 and Table 3 respectively. In these tables, SD_1 stands for single dictionary with averaging weight $w_t = 1$ while SD_β means single dictionary with averaging weight $w_t = (1/f_t)^\beta, \beta = 2.0$. It is similar for PAMD_1 and PAMD_β.

Besides higher PSNR, results by PAMD also have better visual quality than those by SD and the state-of-the-art (e.g., LSSC), including significantly less Moiré Pattern (Figure 2, 3 and 5) and sharper edges (Fig. 4 and 6).

4.3 Discussions

It can be seen that PAMD significantly outperforms single dictionary based approach with 0.46 dB PSNR improvement on average when $w_t = 1$, as shown in Table 1. Moreover, PAMD also outperforms the leading algorithms including CDM [35], RI [36], CAD [3], OAP [4], DL [12] and LPA [27] in non-sparse-coding family, on average with 2.94, 1.98, 1.41, 1.27, 0.52 and 0.08 dB PSNR improvement respectively when $\beta = 2.0$. On several images, PAMD also outperforms the complicated CD [1] and LSSC [10] with respect to both PSNR (Table 2 and 3) and visual quality (Figure 2). Especially, the general artifact by Bayer CFA, Moiré Pattern, is dramatically eliminated.

The success can be partly explained as that patch aware dictionary for each patch class can improve the sparsity level of the representation coefficient vector, i.e., yielding smaller EEF. This is evidenced in Figure 1, which demonstrates that sparser coefficients are accomplished in PAMD framework. As aforementioned, sparser coefficient means that the minimum sample rate required by the compressive theory for good restoration would be lower, then the fixed sample rate 1/3 would become relatively higher. Another interesting finding is that the adaptive weight $w_t = (1/f_t)^\beta$ accomplishes better result than that by $w_t = 1.0$ in PAMD framework while on the contrary in single dictionary framework.

Table 1. Demosaicing results with respect to PSNR (dB). Test images are from the True Color Kodak Images dataset [34]. It can be seen that PAMD framework significantly improves the demosaicing performance over Single Dictionary (SD) based approach, leading algorithms including CDM [35], RI [36], CAD [3], OAP [4], DL [12] and LPA [27]. In this table, the subscript 1 in SD_1 and $PAMD_1$ means averaging weight $w_t = 1$ while β in SD_β and $PAMD_\beta$ means adaptive averaging weight $w_t = (1/f_t)^\beta$. Visual quality comparisons are presented in Fig. 2, 3, 4, 5, and 6. A 15-pixels boarder is excluded for all results.

Img	CDM [35]	RI [36]	CAD [3]	OAP [4]	DL [12]	LPA [27]	SD_1	SD_β	$PAMD_1$	$PAMD_\beta$	$PAMD_1 - SD_1$
01	34.38	35.50	34.91	37.94	38.46	39.45	40.74	40.49	**41.71**	41.69	0.97
02	39.76	39.56	41.14	39.50	40.89	41.36	40.51	40.21	41.39	**41.52**	0.88
03	41.79	41.14	43.36	41.47	42.66	**43.47**	41.52	41.18	42.36	42.17	0.84
04	39.59	40.12	**42.11**	40.00	40.49	40.84	40.70	40.54	40.99	41.18	0.29
05	35.69	36.66	35.57	37.47	38.07	37.51	37.44	37.30	37.91	**37.93**	0.47
06	35.70	38.39	37.72	38.74	40.19	40.92	40.67	40.56	**41.31**	41.29	0.64
07	41.45	41.92	42.11	41.81	42.35	**43.06**	42.01	41.79	42.59	42.62	0.58
08	32.49	34.18	35.12	35.43	35.58	37.13	37.11	36.96	37.43	**37.58**	0.32
09	40.45	41.26	41.14	41.85	43.05	**43.50**	42.13	41.95	42.89	42.93	0.76
10	40.82	41.69	41.14	42.13	42.54	**42.77**	42.29	42.19	42.59	42.58	0.30
11	36.84	38.11	38.59	39.32	40.01	40.51	40.19	40.13	**40.68**	40.64	0.49
12	40.94	42.21	42.11	42.66	43.45	44.01	43.76	43.54	44.34	**44.35**	0.58
13	30.62	31.96	32.95	34.45	34.75	36.08	36.54	36.50	**36.78**	36.75	0.24
14	36.13	36.36	**37.72**	35.70	36.91	36.86	35.45	35.20	36.20	36.51	0.75
15	38.45	38.84	40.35	39.28	39.82	40.09	39.75	39.64	40.27	**40.38**	0.52
16	38.95	42.18	42.11	42.07	43.75	44.02	44.12	43.92	**44.71**	44.66	0.59
17	38.88	40.10	39.68	41.39	41.68	**41.75**	41.37	41.39	41.42	41.35	0.05
18	34.82	35.64	36.67	37.53	37.64	37.59	37.64	37.64	**37.72**	37.69	0.08
19	37.43	39.13	39.10	40.00	41.01	**41.55**	41.18	41.06	41.26	41.47	0.08
20	39.32	39.99	40.35	40.70	41.24	41.48	40.90	40.71	41.38	**41.53**	0.48
21	35.83	37.23	37.72	38.82	39.10	39.61	40.15	40.01	40.65	**40.66**	0.50
22	37.02	37.42	**38.59**	37.67	38.37	38.44	37.77	37.73	38.01	38.06	0.24
23	42.17	41.95	43.36	41.88	43.22	**43.92**	42.31	42.01	42.73	42.75	0.42
24	33.24	34.17	35.34	34.88	**35.55**	35.44	34.87	34.89	34.93	34.83	0.06
Avg	37.61	38.57	39.12	39.28	40.03	40.47	40.05	39.90	40.51	**40.55**	0.46

Table 2. PSNR (dB) comparison with complicated CD [1].

Image	CD [1]	$PAMD_1$	$PAMD_\beta$
01	41.51	**41.71**	41.69
02	41.27	41.39	**41.52**
04	40.86	40.99	**41.18**
07	42.58	42.59	**42.62**
13	36.43	**36.78**	36.75
15	39.77	40.27	**40.38**

Table 3. PSNR (dB) comparison with complicated LSSC [10]. The number of training samples in PAMD for each class is about 4×10^4 while it is more than 2×10^7 for LSSC. Visual quality comparisons can be found in Figure 2.

Image	LSSC [10]	PAMD_1	PAMD_β
01	41.36	**41.71**	41.69
08	37.57	37.43	**37.58**
13	36.35	**36.78**	36.75
21	40.65	40.65	**40.66**

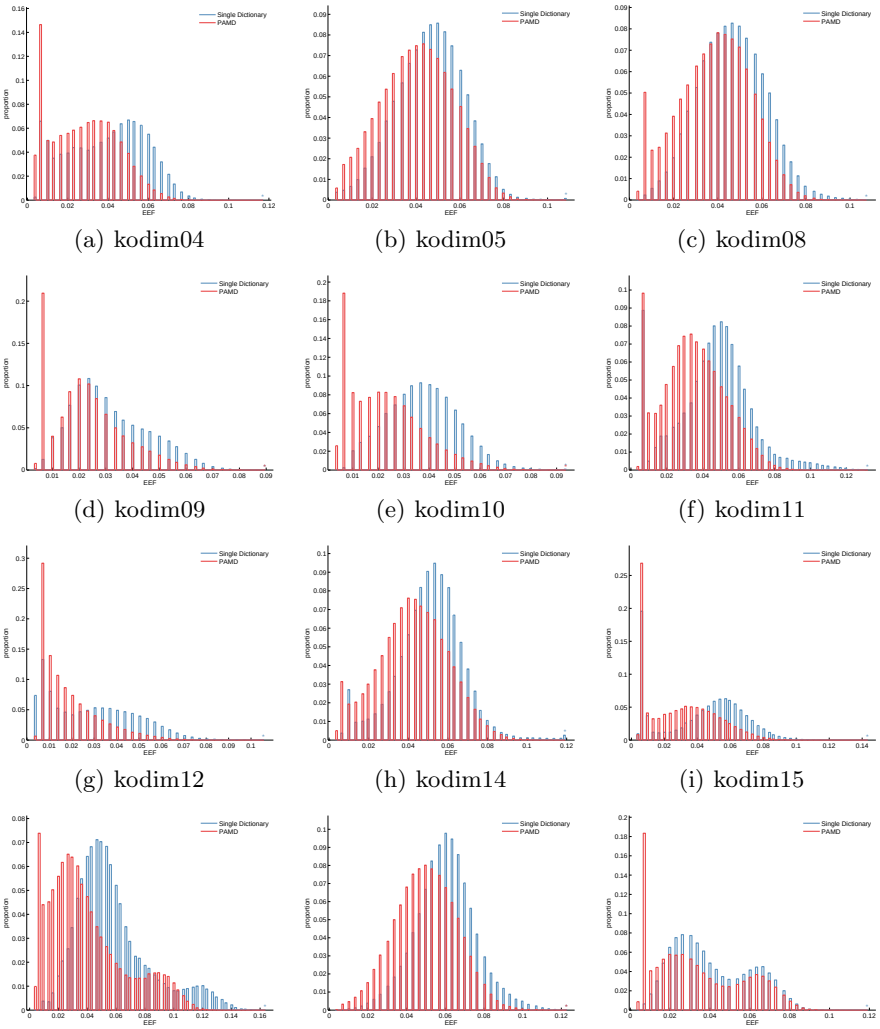
Generally, sparse coding based approaches take longer time than interpolation based methods. Some interpolation based algorithms also take quite a long time such as approximately 1950 seconds per image (768×512) for CDM [35]. Currently, PAMD is implemented serially using MatLab without optimization nor parallel processing. When window steps are 10 pixels (even number, so no need for pre-restoration), PAMD only costs 8 seconds. When the step is 1 pixel, it achieves best restoration with about 700 seconds per image while SD with 500 seconds. Great speed-up can be accomplished by parallel processing.

5 Conclusion

A Patch Aware Multiple Dictionary (PAMD) framework for demosaicing is proposed to improve the sparsity level over dictionaries so that the fixed sample rate would become relatively high, yielding satisfying restoration. Based on the *Energy Exclusiveness Feature* (EEF), PAMD framework classifies patches in both training and testing stages. A dictionary would be trained for each class; and candidate atoms for sparse coding would be constrained within a class-specific dictionary so that low atom correlations are ensured. Multiple dictionaries pack more prior knowledge from a large number of training samples than single dictionary. As evidenced by our extensive experiments, class-specific dictionaries in PAMD framework significantly improve the sparsity level of patches and accomplish much better results than single dictionary based approach, with respect to both PSNR and visual quality. Especially, the general artifact by Bayer CFA, Moiré Pattern, is dramatically eliminated. PAMD also significantly outperforms many leading algorithms and outshines the complicated state-of-the-arts on several images. PAMD is very simple and can corporate with the existing state-of-the-arts. Particularly, based on patch classification, existing sparse coding based algorithms can be performed in each class. Moreover, sparser representation is an advantage for many sparse coding based applications to which PAMD is potentially able to be applied.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grants 61272232 and MOST under Grants 2012AA011602.



(a) kodim04 (b) kodim05 (c) kodim08

(d) kodim09 (e) kodim10 (f) kodim11

(g) kodim12 (h) kodim14 (i) kodim15

(j) kodim17 (k) kodim18 (l) kodim19

Fig. 1. Histograms of EEFs for patches in several test images. For each image, 96393 patches are used for building the histogram. The horizonal axis is the EEF f. The smaller f, the sparser the representation coefficient vector is. The vertical axis is the proportion (normalized, between 0 and 1) of patches whose EEFs fall in certain bin. The blue one is for Single Dictionary (SD) while the red one is for PAMD framework. Note that the red and blue histograms have the same total area (1.0). It can be seen that significantly sparser coefficients are accomplished in PAMD framework. Visually, the red area prevails in the left area (small f) while the blue area prevails in the right area (large f). It partly explains the success of PAMD.

(a) Kodim08 ground truth image. The cropped part is highlighted in the green rectangle.

(b) Cropped ground truth

(c) Result by our PAMD_{β}

(d) Result by SD_1

(e) Result by OAP [4] which is a leading algorithm based on interpolation.

(f) Result by LSSC [10] which is the state-of-the-art based on sparse coding.

Fig. 2. Visual quality comparison between our PAMD, SD (Single Dictionary), OAP [4] and LSSC [10]. OAP is one of the leading algorithms in interpolation family. LSSC is the state-of-the-art and sparse coding based, which involves an online dictionary learning algorithm, more than 10^6 training patches, clustering, on-the-fly training from testing image patches, and group sparsity constrain techniques. LSSC is complicated and requires heavy computation overhead in demosaicing stage. PAMD is very simple and the average number of training samples for each class is only about 4×10^4, with low computation overhead in testing stage. It can be seen that by our PAMD framework, **the general artifact by Bayer CFA, Moiré Pattern, is dramatically eliminated**.

(a) Kodim01 ground truth image. The cropped part is highlighted in the green rectangle.

(b) Cropped ground truth

(c) Result by our PAMD$_\beta$

(d) Result by SD$_1$

Fig. 3. Visual quality comparison. PAMD greatly reduces **Moiré Pattern artifact**.

(a) Kodim03 ground truth image. The cropped part is highlighted in the green rectangle.

(b) Cropped ground truth

(c) Result by our PAMD$_\beta$

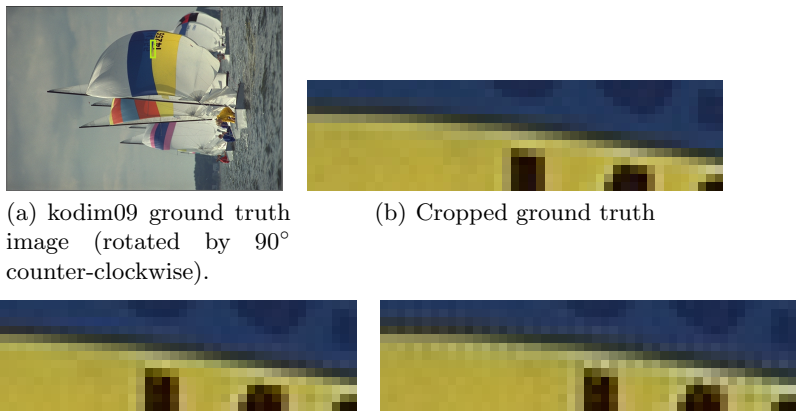
(d) Result by SD$_1$

Fig. 4. Visual quality comparison. PAMD achieves **sharper edges**.

(a) kodim01 ground truth image. (b) Cropped ground truth

(c) Result by our PAMD$_\beta$ (d) Result by SD$_1$

Fig. 5. Visual quality comparison. PAMD greatly reduces **Moiré Pattern artifact**.



(a) kodim09 ground truth image (rotated by 90° counter-clockwise). (b) Cropped ground truth

(c) Result by PAMD$_\beta$ (d) Result by SD$_1$

Fig. 6. Visual quality comparison. PAMD achieves **sharper edges**.

References

1. Moghadam, A., Aghagolzadeh, M., Kumar, M., Radha, H.: Compressive framework for demosaicing of natural images. Image Processing, IEEE Transactions on **22** (2013) 2356–2371
2. Shao, L., Rehman, A.U.: Image demosaicing using content and colour-correlation analysis. Signal Processing (2013)
3. Ur Rehman, A., Shao, L.: Classification-based de-mosaicing for digital cameras. Neurocomputing **83** (2012) 222–228
4. Lu, Y., Karzand, M., Vetterli, M.: Demosaicking by alternating projections: Theory and fast one-step implementation. Image Processing, IEEE Transactions on **19** (2010) 2085–2098
5. Gunturk, B., Glotzbach, J., Altunbasak, Y., Schafer, R., Mersereau, R.: Demosaicking: color filter array interpolation. Signal Processing Magazine, IEEE **22** (2005) 44–54
6. Gunturk, B., Altunbasak, Y., Mersereau, R.: Color plane interpolation using alternating projections. Image Processing, IEEE Transactions on **11** (2002) 997–1013
7. Menon, D., Calvagno, G.: Regularization approaches to demosaicking. Image Processing, IEEE Transactions on **18** (2009) 2209–2220
8. Ferradans, S., Bertalmío, M., Caselles, V.: Geometry-based demosaicking. Image Processing, IEEE Transactions on **18** (2009) 665–670
9. Malvar, H.S., He, L.w., Cutler, R.: High-quality linear interpolation for demosaicing of bayer-patterned color images. In: Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on. Volume 3., IEEE (2004) iii–485
10. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Non-local sparse models for image restoration. In: Computer Vision, 2009 IEEE 12th International Conference on. (2009) 2272–2279
11. Mairal, J., Elad, M., Sapiro, G.: Sparse representation for color image restoration. Image Processing, IEEE Transactions on **17** (2008) 53–69
12. Zhang, D., Wu, X.: Color demosaicking via directional linear minimum mean square-error estimation. Image Processing, IEEE Transactions on **14** (2005) 2167–2178
13. Chung, K.H., Chan, Y.H.: Color demosaicing using variance of color differences. Image Processing, IEEE Transactions on **15** (2006) 2944–2955
14. Tanaka, M., Okutomi, M.: Color kernel regression for robust direct upsampling from raw data of general color filter array. In: Proceedings of the 10th Asian Conference on Computer Vision - Volume Part III. ACCV'10, Berlin, Heidelberg, Springer-Verlag (2011) 290–301
15. Candes, E., Romberg, J., Tao, T.: Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. Information Theory, IEEE Transactions on **52** (2006) 489–509
16. Candes, E., Wakin, M.: An introduction to compressive sampling. Signal Processing Magazine, IEEE **25** (2008) 21–30
17. Zhang, M., Tao, L.: Volume reconstruction for mri. In: Pattern Recognition (ICPR), 2014 22nd International Conference on. ICPR'14, IEEE (2014) 3351–3356
18. Mohimani, H., Babaie-Zadeh, M., Jutten, C.: A fast approach for overcomplete sparse decomposition based on smoothed ℓ^0 norm. Signal Processing, IEEE Transactions on **57** (2009) 289–301

19. Tropp, J., Gilbert, A.: Signal recovery from random measurements via orthogonal matching pursuit. Information Theory, IEEE Transactions on **53** (2007) 4655–4666
20. Candes, E., Romberg, J.: l1-magic: Recovery of sparse signals via convex programming. URL: www.acm.caltech.edu/l1magic/downloads/l1magic.pdf (2005)
21. Lee, H., Battle, A., Raina, R., Ng, A.Y.: Efficient sparse coding algorithms. In Schölkopf, B., Platt, J., Hoffman, T., eds.: Advances in Neural Information Processing Systems 19. MIT Press (2007) 801–808
22. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society. Series B (Methodological) (1996) 267–288
23. Aharon, M., Elad, M., Bruckstein, A.: k-svd: An algorithm for designing overcomplete dictionaries for sparse representation. Signal Processing, IEEE Transactions on **54** (2006) 4311–4322
24. Dhillon, I.S., Sra, S.: Generalized nonnegative matrix approximations with bregman divergences. In: In: Neural Information Proc. Systems. (2005) 283–290
25. Bach, F.R., Jordan, M.I.: Kernel independent component analysis. J. Mach. Learn. Res. **3** (2003) 1–48
26. Filipovic, M., Kopriva, I., Cichocki, A.: Inpainting color images in learned dictionary. In: Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European, IEEE (2012) 66–70
27. Paliy, D., Katkovnik, V., Bilcu, R., Alenius, S., Egiazarian, K.: Spatially adaptive color filter array interpolation for noiseless and noisy data. International Journal of Imaging Systems and Technology **17** (2007) 105–122
28. Menon, D., Calvagno, G.: Regularization approaches to demosaicking. Image Processing, IEEE Transactions on **18** (2009) 2209–2220
29. Li, X., Gunturk, B., Zhang, L.: Image demosaicing: A systematic survey. In: Electronic Imaging 2008, International Society for Optics and Photonics (2008) 68221J–68221J
30. Donoho, D.L.: Compressed sensing. Information Theory, IEEE Transactions on **52** (2006) 1289–1306
31. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: ICML. (2009) 87
32. Olmos, A., et al.: A biologically inspired algorithm for the recovery of shading and reflectance images. Perception **33** (2003) 1463–1473
33. Hyvärinen, A., Oja, E.: A fast fixed-point algorithm for independent component analysis. Neural computation **9** (1997) 1483–1492
34. Kodak: Kodak lossless true color image suite (2004) [Online available] `http://r0k.us/graphics/kodak`.
35. Zhang, L., Wu, X., Buades, A., Li, X.: Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. Journal of Electronic Imaging **20** (2011) 023016–023016–16
36. Kiku, D., Monno, Y., Tanaka, M., Okutomi, M.: Residual interpolation for color image demosaicking. In: IEEE International Conference on Image Processing, ICIP 2013, Melbourne, Australia, September 15-18, 2013. (2013) 2304–2308