



HAL
open science

Demosaicing using Dual Layer Feedforward Neural Network

Prakhar Amba, David Alleysson, Martial Mermillod

► **To cite this version:**

Prakhar Amba, David Alleysson, Martial Mermillod. Demosaicing using Dual Layer Feedforward Neural Network. CIC 2018, Nov 2018, Vancouver, Canada. hal-02106659

HAL Id: hal-02106659

<https://hal.science/hal-02106659v1>

Submitted on 23 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Demosaicing using Dual Layer Feedforward Neural Network

Prakhar Amba¹, David Alleysson¹, Martial Mermillod^{1,2}

¹ Laboratoire de Psychologie et Neurocognition, Univ. Grenoble Alpes, France

² Laboratoire Jean Kuntzmann, Univ. Grenoble Alpes, France

Abstract

Color demosaicing is the problem of recovering full color/spectral channel from a subsampled image captured by single-chip digital cameras covered by Color Filter Array (CFA). Several algorithms have been proposed in the literature, however most of them are tuned to a particular arrangement of color filters. In this paper, we propose a generic algorithm based on a simple Neural Network (NN) architecture, which is trained on a small image database and which gives competitive results. To prove our statement, we test our network on several state of art CFAs and a 5 channel Spectral Filter Array (SFA). We demonstrate our result both on simulated images coming from standard image databases and also RAW images for a 5 channel SFA camera.

Introduction

To capture color images, digital cameras overlay a mosaic of Red, Green and Blue filters on the sensor. From this sub-sampled image one needs to estimate the missing colors at each pixel position to recover the full color image. This reconstruction step is known as demosaicing. Bayer [10] is the most common Color Filter Array (CFA) arrangement used in the industry for digital cameras.

Recently there is a growing interest in development of Spectral Filter Arrays (SFAs), going beyond three color filters and even adding Near-Infrared filter on the same mosaic. IMEC has proposed SFAs with 25 color channels [3] and Silios [5] with 9, are commercial propositions. SFAs give us greater flexibility in choosing the filter spectral sensitivities depending on application. SFAs open up new applications in computer vision, accurate color reproduction, spectral reconstruction, detection of spectral signature of objects, etc. Several demosaicing algorithms developed for CFAs have been extended to the SFAs. Therefore it is important to consider it to be a generic problem. Still demosaicing for SFAs is a new challenge because unlike CFAs the sampling can be even more sparser and the inter-correlation between pixels is not so high. Unlike RGB CFAs, where generally filters are wide-band filters with high correlation. Now, we no longer have that limitation as we can have filters with narrow-band sensitivities [35].

In the state of the art, Bayer is most popular CFA and due to availability of source code for several demosaicing algorithms and standardized image databases like Kodak [4], it is easy to benchmark with. Due to periodic nature of the Bayer CFA, simple demosaicing algorithms like bilinear give rise to false colors and moire in recovered images. Edge directed algorithms have been proposed to counter that. They interpolate along the contours and not across them. [14, 15, 18, 21, 23, 24, 32, 36, 41, 42] These algorithms are computationally more expensive than spatial/frequency selection based methods. Different arrangement of

CFAs have been proposed in literature which optimizes the distribution of luminance and chrominance in the frequency domain and make designing of reconstruction filters more accurate [7]. However, frequency selection based demosaicing cannot be used for all CFAs especially those with random arrangement. LMMSE (linear minimum mean square error) based algorithms have been proposed which can be used to demosaic random CFAs [8, 11]. Authors consider demosaicing to be an inverse problem of estimating a linear operator which inverses the effect of mosaic by learning its weights on an image database. As the size of the recovered image is larger than the size of CFA size, the estimated solution is not very good. So, the linear operator can be further stabilized by considering an area of neighboring pixels. LMMSE was further extended to the SFA domain [9]. The authors demonstrated that they could gain resolution and less false colors at the expense of artifacts for a prototype SFA camera [35] compared to the Binary-Tree approach [27]. In the present paper, we hypothesize that a linear solution is not enough due to increased sparsity and reduced spectral correlation.

A non linear solution extends the degree of freedom for the weights and therefore can give potentially better results. Neural Networks (NN) have been shown to be good candidate for such problems and machine learning based solutions are already used extensively for image classification/recognition, pattern matching, etc. [19]. The idea being to train a neural network to recognize patterns on a huge database.

Recently, a three layer neural network using deep learning for demosaicing [37] was proposed. Authors trained their network on 2992 down-sampled images from Flickr [1]. Deep residual network has been proposed for joint demosaicing and super-resolution [43]. Also in [13], authors have proposed a neural network based on convolutional neural networks having 15 layers, training them on more than a million images. They use sRGB images, which are down-sampled using bicubic interpolation to avoid incorporating the distortions caused by the image processing pipeline. They demonstrate that their network generalizes well to linear data (directly from RAW images, so there is no gamma, post processing, compression, etc.) also. Probably as the network trains on millions of images, the network is very generic and therefore has averaged out the differences in image processing pipeline, sRGB transform, noise removal, sharpening, jpg compression, etc. A 15 layer neural network might not be suitable for embedded system for which SFAs are destined as they have applications in machine vision where real time performance, cost and energy consumption considerations are paramount.

Moreover, all these approaches train the network on already demosaiced images. It is known that for deep learning networks, large datasets for training are beneficial. Image databases which provide ground truth images for SFA filters (possible using a

color-wheel camera system having similar filters) may not be available. We have image databases in reflectance domain like Finlayson [12, 17], Cave [39] and SCIEN [33], which can be used to simulate color images in any filter domain. But these databases are usually small in size. Therefore it is important to consider possibility of a neural network approach which doesn't require large training dataset. Ideally, one needs large database of ground truth images available. However, this is not the case for hyperspectral databases.

In this paper, we propose a simple dual layer Neural Network for demosaicing, which can be trained for demosaicing any random CFA/SFAs. We restrict the study on this simple network to allow a fair comparison with LMMSE methods which apply favorably on embedded systems with reduced computational power. To demonstrate the capabilities of our Network, we present results on three different studies, namely RGB and RGBW (where 'W' or white pixel is linear combination of RGB) CFAs, Monno 5 channel CFA [30, 31] and 5 channel prototype SFA camera [29], refer to Figure 2. For the purpose of this paper, we are concerned only with spatial reconstruction from the subsampled filter space and not spectral reconstruction which is beyond the scope of this paper.

Neural Network Configuration

Let us define a 'super-pixel' to be the basis pattern [34], the most basic pattern of mosaic of filters which is repeated across the surface of the sensor forming the Color Filter Array. For the Bayer [R G; G B], the 'super-pixel' is of size 2×2 pixels. For Fuji XTrans CFA [2], the super-pixel is of size 6×6 . Let h and w be the number of rows and columns respectively in the super-pixel. We consider demosaicing problem to be of block shift invariant, same solution is proposed for each super pixel pattern. Now for $h \times w$ pixels in 'super-pixel', the goal of demosaicing is to estimate P_{hw} pixels of color image, where P is number of color channels. Now P can be 3 for RGB CFAs or more than 3 when dealing with more filters. The way we test demosaicing with neural network is the following. From true color images taken in the database, we simulate the CFA sampling by suppressing corresponding colors. We then input the CFA images to the network and compute the estimated weights and bias of the neurons to reconstruct color image at the output. The criterion for a good reconstruction is the Mean Square Error (MSE) between the output of the neural network and the original true color image. We take each 'super-pixel' and decompose it into a column vector when presenting it to the network. It is easy to understand that the solution at the moment will not be very stable as the size of output is greater than inputs. Therefore, we consider a neighborhood of W_h pixels and W_w pixels encapsulating the super-pixel. Thereby, we increase the size of input to $W_h \times W_w$ while presenting it to our network.

We used a two layer feedforward neural network (Figure 1) of fitting type to solve the demosaicing problem. The first layer is composed of n sigmoid neurons and the second layer of P_{hw} linear neurons. We choose this particular configuration as we consider demosaicing to be a data fitting problem and not a pattern matching problem.

First, we train this network on the Kodak database [4] and compare results with the state of art for Bayer, Fuji, RGB+panchromatic CFAs and other optimized CFAs proposed in the literature. Here, the white (panchromatic) pixel is simu-

lated as the linear combination of RGB pixels. For comparison with LMMSE algorithm [8], we considered the 4×4 #2 and CNRS CFA. Refer to Figure 2 for visualization of the CFAs. Testing of demosaicing algorithms on RGB CFAs simulated on the Kodak database is the most extensively studied problem in the literature and therefore helps in establishing our algorithm in comparison with other.

Secondly, we train the neural network on 5-band TokyoTech multispectral image database **5-BAND Tokyo**¹ and use it to demosaic a 5 color CFA (see Figure 2) [30, 31]. This database is a true-color image database like the Kodak database, however instead of 3 it has 5 color channels.

Finally, we train the neural network on images from 31-band Tokyotech Multispectral database **31-BAND Tokyo**². This is a hyperspectral image database with images from 420nm to 720nm with images every 10nm. Therefore we need to first render images using filter spectral sensitivity of prototype camera [29] and use them to train our network. Similarly, we could also train our neural network on rendered images from Finlayson or Cave hyperspectral database and use this to demosaic RAW images. To note the filter sensitivities here are not same as those for the 5-Band dataset. Therefore it cannot be directly used. Demosaicing exercise on the 5-Band dataset serves more to compare demosaicing algorithms as ground-truth images are available. The authors have already shown their approach using Weighted Guided Filter (WGF) [29] to be better than Binary-Tree Edge Sensing algorithm [25–27] therefore we choose not to repeat results for Binary Tree. For this camera, we present results both on rendered images from hyperspectral image databases and also on real RAW images captured with this camera. Simulation on rendered images allow us to measure metrics like PSNR, as ground truth is available which allows quantitative comparison.

We trained the neural network using Matlab's neural networking toolbox. The training is not done on images but rather on vectors composed of super-pixel and neighborhood window. For the 24 images of size 768×512 in Kodak database, this represents $(768/2) \times (512/2) \times 24 = 2.3$ millions different vectors on a whole for the Bayer CFA of size 2×2 . For RGB database, we used 70% of data (of super-pixel vectors) for training of network, 15% was used for validation and 15% for testing. Data was divided randomly. This is the parameter used by default by Matlab. This splitting of data is not by images but by vectors composed of super-pixels with their neighborhood window. Therefore, is it not possible to detail composition of each set. In terms of PSNR, the result for testing on only 15% of vectors (unused for training) and full 100% is very similar. When we present results, we test 100% of images as in the state of art results are provided by testing on 100% of images therefore we choose to do so to allow comparison. As the training time is quite long, it is not possible to do a complete leave one out testing. For the 5-band TokyoTech database use for the Monno5ch CFA, we use only 6.25% of super-pixels vectors for training and again test full 100% of images. For the 31-band TokyoTech multispectral image database, Finlayson and Cave, used for the Monno5ch SFA, we used only 25% of vectors for training. Despite the small percentage of input data,

¹<http://www.ok.sc.e.titech.ac.jp/res/MSI/MSIdata.html>

²<http://www.ok.sc.e.titech.ac.jp/res/MSI/MSIdata31.html>

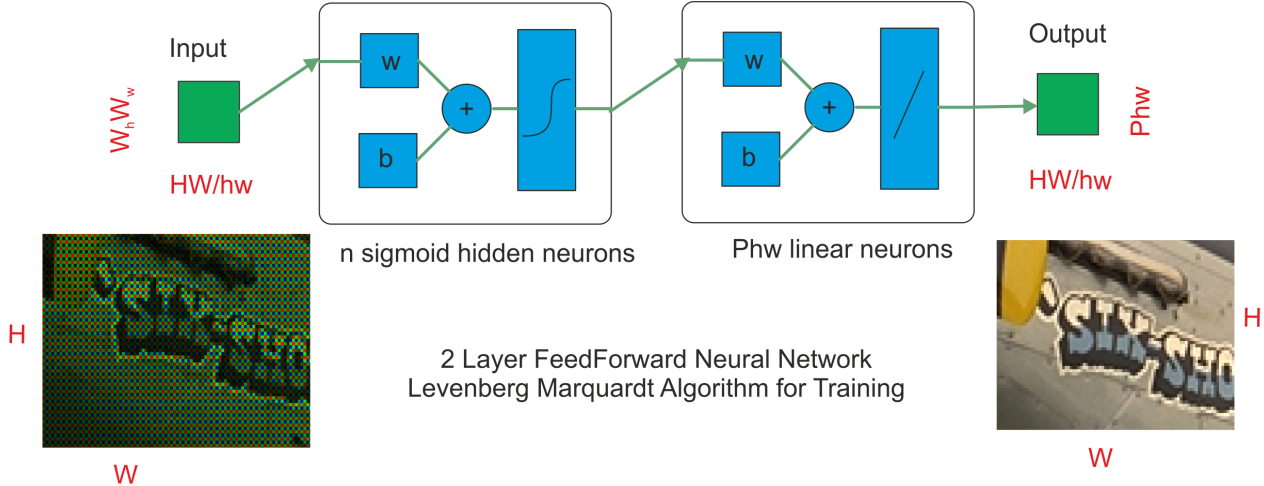


Figure 1. Neural Network model used. We used Matlab’s Neural Networking toolbox to train our network. A neighborhood window of size $W_h \times W_w$ is considered around each super-pixel of size h_w . We need to replace P by P_s for SFAs.

the network performs well. We trained the network using scaled conjugate gradient method with back-propagation.

Results

Testing for RGB and RGBW CFAs

We trained the neural network by paralleling the task on both CPU and GPU. The system consisted of Intel i7 6700K with Nvidia GTX 1080. After training the network based on procedure described earlier, we tested the network for the entire Kodak image database. For achieving a good PSNR, we found there are two factors, having a big neighborhood size and consequently more neurons in the hidden layer number 1. The training time also depends on the number of neurons. Increasing the number of neurons greater than the size of input gives diminishing result in performance with regard to computational time increases and memory requirement, which is known as over-fitting problem. Table 1 and 2 shows the result for testing our NN approach on the Kodak [4] and McM [42] image database respectively. All the values reported are averages across the image database. For instance, μ is average of Color Peak Signal to Noise Ratio (CPSNR) for 24 images. σ_{rgb} tells us the variance of PSNR between the RGB channels. A lower value indicates that all three color channels are equally well reconstructed. As expected for CNRS and 4x4 #2, we have a lower σ_{rgb} as the CFAs have quasi-equal distribution of color filters unlike Bayer or Fuji, where green channel is better reconstructed compared to other two. σ denotes the variance of PSNR across all the images across the database. Therefore, a lower value indicates that all the images are well reconstructed. Figure 3 shows the fence region of the Lighthouse image demosaiced using our algorithm on the different CFAs. CFAs like CNRS (to some extent) and 4x4 #2 avoid false colors and moire due to random arrangement in the mosaic compared to the Bayer CFA. Execution times reported are for reconstruction on CPU only.

In Table 1, the Bayer CFA, a window size ($W_h W_w$) of (10,10) was used with 100 neurons. The training took 20.6 hours giving a performance of 40.71dB. Using only 40 neurons with ($W_h W_w$)

Table 1: Performance of Ours NN, expressed as averages across images in Kodak database

CFA	μ	SSIM	ΔE	σ_{rgb}	σ	time(s)
Bayer	40.71	0.9930	1.25	4.39	5.17	0.15
Fuji	39.10	0.9912	1.53	3.42	5.61	0.15
CNRS	40.01	0.9928	1.41	0.58	5.85	0.15
4x4 #2	41.00	0.9940	1.31	0.77	5.47	0.22
RGBW	40.54	0.9932	1.47	1.25	5.20	0.20
Kodak 2.0	38.82	0.9905	1.76	2.05	5.46	0.20
SonyRGBW	38.11	0.9891	1.86	2.80	5.20	0.20

Table 2: Performance of Ours NN, expressed as averages across images in McM

CFA	μ	SSIM	ΔE	σ_{rgb}	σ	time(s)
Bayer	36.96	0.9865	2.90	6.50	7.64	0.12
Fuji	35.45	0.9821	3.48	7.03	8.22	0.12
CNRS	35.61	0.9837	3.60	1.80	8.62	0.12
4x4 #2	36.42	0.9863	3.32	2.01	8.60	0.12
RGBW	36.88	0.9870	3.28	1.97	8.11	0.15
Kodak 2.0	35.26	0.9820	4.11	1.45	8.48	0.12
Sony RGBW	35.08	0.9812	4.14	1.50	8.46	0.12

(11,11) gave us 40.19dB in training time of 7.6 hours. Despite having less neurons in the hidden layer compared to number of inputs, the result is still good. For comparison [13], reported a training time of 2-3 weeks on a much faster GPU. For Fuji, we used ($W_h W_w$) of (15,15) with 225 neurons. For 4x4 #2 ($W_h W_w$) of (13,13), with 40 neurons we have 40.32dB, 100 neurons 40.70dB, 169 neurons 41dB. Unlike LMMSE where the numbers of weights in the demosaicing operator is equal to the size of input, here the relation between number of neurons and size of input is not so straightforward. Despite having less neurons in the hidden layer compared to number of inputs, the result is still good. Depending on application we can reduce the number of neurons / neighborhood to slightly reduce Average PSNR in order to gain computational speed.

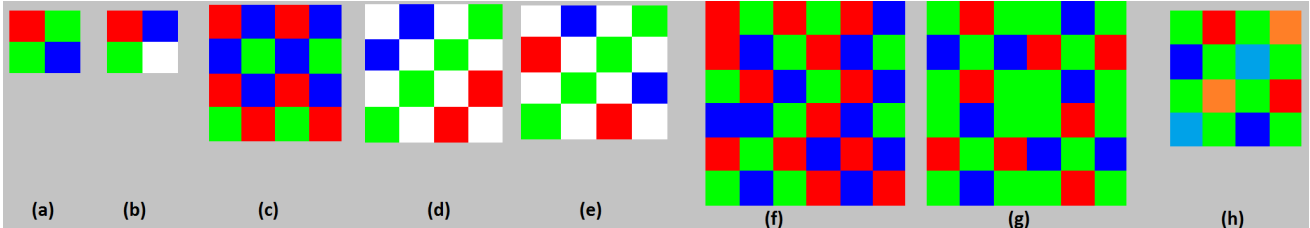


Figure 2. CFAs and SFA tested. (a) Bayer [10], (b) RGBW, (c) 4x4 # 2 [8], (d) Kodak 2.0 [20], (e) Sony RGBW [16], (f) CNRS [6], (g) Fuji [2], (h) Monno 5ch CFA [30,31] (h) Monno 5ch SFA [29]. Note for Monno5ch, the same pattern is referred as CFA when using spectral sensitivities as defined in [30,31], tested on the 5-band TokyoTech database which incorporates these same sensitivities. Referred to as SFA when using spectral sensitivities as defined in [29], so we have RAW images coming from real prototype camera.

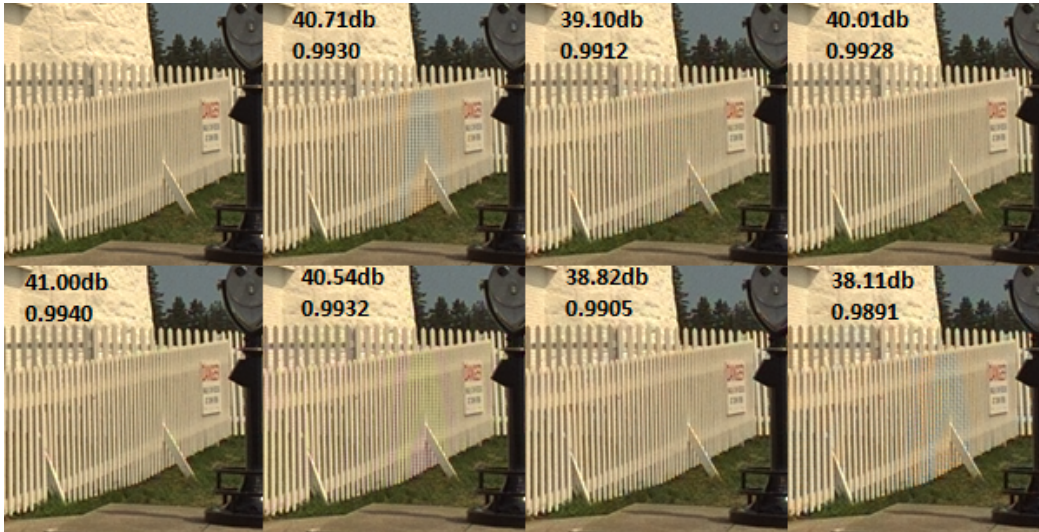


Figure 3. Lighthouse Image crop. Demosaiced using our NN method. (Top Row: Left to Right) Original, Bayer, Fuji, CNRS. (Bottom Row: Left to Right) 4x4 # 2, RGBW, Kodak 2.0, Sony RGBW. On the image are presented values of average PSNR (μ) and average SSIM for the entire Kodak database.

Table 3: Result of average CPSNR for Kodak database for our method compared with state of art. LMMSE [8] reported values of CPSNR, which were not clipped between [0 1]. Here we clip all results. Gharbi [13] is a NN approach using deep learning. Another Neural Network based approach [37] reports RMSE values, equivalent value for μ is 37.18 for 19 images only.

CFA	Average PSNR μ			
	Ours	LMMSE [8]	ACUDE [40]	Gharbi [13]
Bayer	40.71	39.13	40.71	41.2
Fuji	39.10	39.03	39.54	
CNRS	40.01	40.03		
4x4 #2	41.00	40.68		
RGBW	40.54	39.74		
Kodak 2.0	38.82	38.43	38.70	
SonyRGBW	38.11	37.38	38.10	

Table 3, 4 shows the result for Average PSNR for our method compared with the state of art. Our Neural Network gives a peak performance of 40.71dB using 2 layers only compared to 41.2 dB for the 15 layer Gharbi’s Neural Network for the Kodak database. It outperforms LMMSE in PSNR. But it is slower in execution speed. Compared to ACUDE, it is slightly better for RGB plus panchromatic CFAs, like Kodak 2.0 and Sony RGBW CFA. We

Table 4: Result of average CPSNR for McM database for our method compared with state of art. LMMSE [8] reported values of CPSNR, which were not clipped between [0 1]. Here we clip all results. LLSC for Bayer gives 36.14

CFA	Average PSNR μ			
	Ours	LMMSE [8]	ACUDE [40]	Gharbi [13]
Bayer	36.96	35.70	36.38	39.5
Fuji	35.45	35.22	35.99	
CNRS	35.61	35.48		
4x4 #2	36.42	36.04		
RGBW	36.88	35.77		
Kodak 2.0	35.26	34.62	35.15	
SonyRGBW	35.08	34.35	34.87	

find similar trends for the McM database, with the Gharbi’s NN having an even bigger advantage.

LMMSE is a linear solution and it take 0.10s/image. Gharbi NN [13] report 2.9s per Mpixel which corresponds to roughly 1.14s for a Kodak image. However, they use a slower processor which has 14% lower clock speed. So, we can roughly extrapolate it to 0.98s per image. For sure they use 15 layer NN while we are using only 2 layers, take 0.15s per image. For ACUDE, with the code publicly available, it takes approximately 1 hour to process

Table 5: Result of performance metrics as averages across 5 channels for our method compared with state of art for Monno5ch CFA for 5-band multispectral TokyoTech dataset.

	Monno5ch CFA TokyoTech 5-band			
	Ours	LMMSE [8]	MSRI [28]	WGF [29]
PNSR	44.72	45.16	44.45	43.11
SSIM	0.9946	0.9945	0.9942	0.9923
σ_{5ch}	9.74	12.91	9.74	8.50
σ	9.40	12.40	17.50	16.30
Time(s)	0.78	0.46	14.63	31.21

a single image with Bayer CFA. However, the authors claim on their website that they can process in under 1s per image.

For Bayer, despite having PSNR higher than 40dB, both our NN and ACUDE exhibit false colors in the fence part of Lighthouse image, see Figure 3. Gharbi et al. don't present the result. Only LLSC [22] avoids that, it has an average PSNR of 41.46dB, however it takes approx 6 minutes per image (on xeon e5 1620, note the compiled code is single threaded, so there is potential to improve the timing). We do not know how to adapt their algorithm for CFAs other than Bayer. Therefore, we do not present the result in the Table 3 & 4. It is clear by looking at Figure 3, that PSNR is not enough to quantify false colors and artifacts. As most demosaicing papers present results in PSNR, it remains a metric of choice.

Monno5ch CFA with images from 5-band multispectral dataset

For the Monno5ch CFA (see Figure 2), we trained the NN on the 5-band TokyoTech multispectral dataset. These 5 channels in this database already incorporate the filter sensitivities of this CFA [30]. We have about 147 million pixels of color data available. Training on the entire dataset would be very long. Therefore we selected the first 6.25% of pixels from the database for training purpose. We report results on full 100% of data. Finally, we report results for average PSNR and other metrics for all the images for the 5 bands, see Table 5. We achieved an average CPSNR (5 channel) of 44.72dB for a window size of 10, using 100 neurons. It took 1.77 hours to train this network. Increasing the number of neurons or neighborhood size didn't yield better results. Training over entire database might give better results. We are limited by memory considerations as of now. We found that LMMSE has the best performance for this dataset in terms of Average CP-SNR. Figure 4 show crops of demosaiced images converted from 5 channels to sRGB for the proposed NN, LMMSE and WGF method. The demosaiced images have 5 channels. For displaying them we first need to convert them to sRGB domain for which we used the transform based on linear spectral reflectance estimation and then using the color matching function (XYZ) and then conversion from XYZ to sRGB as used by the authors in [29]. It is difficult to visually make out much difference between the three. It seems MSRI has a slight advantage, less false colors in reflection of light in the toy car for instance. In terms of execution speed, our NN and LMMSE are several magnitudes faster than MSRI or WGF.

Testing for Spectral Filter Arrays: Monno5ch SFA with RAW images captured from actual camera

Further Monno et al. implemented the SFA physically [29] and they have provided the spectral sensitivity of the realized filters and RAW images obtained from this camera system. Now the sensitivities of these physical filters differ from 5-band multispectral database which was used earlier. NN trained on this dataset is no longer useful. They have also shared a 31-band multispectral image database. Similar hyperspectral image databases like Finlayson [12] and Cave [38] are available. We can render full resolution 5 channel images using filter sensitivities for this camera using the reflectance image formation model. Basically, the full color image Y is the product of illuminant L multiplied by scene reflectance Z , rendered by the filter sensitivities F_{P_s} , $Y = F_{P_s}LZ$, where P_s is number of spectral channels. Now this full color image is subsampled by the mosaic M to give the CFA/SFA image X . Now for the NN training, X is used as input and Y as output. We generate X and Y from a hyperspectral image database using M and F as determined from the SFA.

Table 6 shows the evaluation of metrics on rendered images from the Finlayson, Cave and Tokyotech 31-band hyperspectral database. We use only WGF method for evaluation as this is what the authors use to present results from RAW camera and not MSRI [29]. The values are for 5 channel demosaiced images, no post processing like gamma is applied here. We find that NN outperforms the rest in 2 out of 3 image databases. Also the WGF method has very high σ , variance of PSNR across all images, which implies that some of the images are very well reconstructed while others not so much.

Figure 5, shows the comparison of three approaches on RAW image captured by this camera. With Ours and LMMSE, the output image is sharper with less false colors but artifacts are presents. A textured pattern is present in flat regions of the scene. A filter to sRGB space transform as described earlier was applied for displaying colour images. To demonstrate generality of our training and method, we show results of demosaicing RAW images where learning was performed on Finlayson hyperspectral database. We have similar results if training is done on Cave or TokyoTech-31 dataset.

Discussion

For the Kodak image database, our neural network is mostly better than LMMSE approach, which is probably because being non linear it had more degree of freedom to adapt to the training data. This come at the cost of being slightly slower. It is still several magnitudes faster than other state of art algorithms like ACUDE or the NN approach by Gharbi [13]. Unlike Gharbi which gains 0.5dB on PSNR, our approach doesn't need to train on millions of images while being 6.5 times as fast in demosaicing. Both still have false colors in high frequency areas for Bayer CFA. Our network uses only 2 layers and yet it is so close to performance to Gharbi's 15 layer CNN, is due to better coding of the input and output vectors, where we use a neighborhood window around each 'super-pixel' to augment the input vectors. For random CFAs like CNRS and 4x4 #2, we don't have false colors with our NN approach. Compared to other state of art algorithms, it is generally better, specially for CFAs with white pixels.

Comparing for the Monno 5 channel CFA, with images coming from 5-band multispectral dataset [30], in terms of PSNR,



Figure 4. Demosaiced Image from 5-band TokyoTech set data using Monno5ch CFA. Left to Right: Ours NN, LMMSE, MSRI

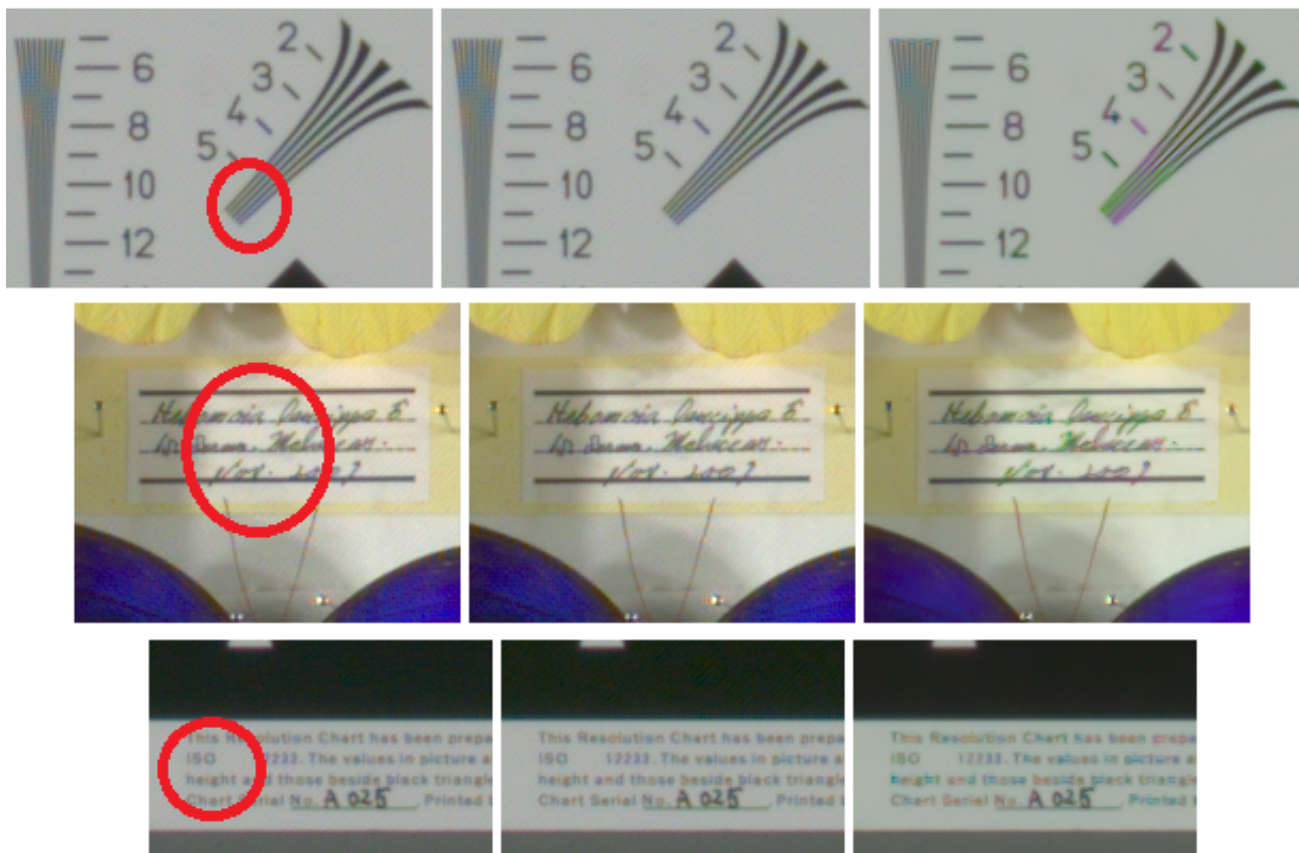


Figure 5. Demosaiced RAW Image from Monno5ch SFA camera. Left to Right: Ours NN, LMMSE, WGF. Notice that text is sharper with NN and LMMSE, there are less false colors but there are artifacts in flat regions.

LMMSE gives the best performance. Visually it is difficult to evaluate, however it appears that MSRI has a very slight advan-

tage over false colors. This comes at a cost of execution time, MSRI being considerably slower. Now one of the limitations with

Table 6: Results for our method compared with state of art for Monno5ch SFA with filter sensitivities simulated on hyperspectral database. SFA images are simulated on rendered images from these databases and reconstructed. For NN and LMMSE, same database is used for training and reconstruction. Illuminant is emuda5 [29] as measured.

Monno5ch SFA RAW camera				
Database		Ours	LMMSE [8]	WGF [29]
Finlayson	PNSR	41.63	41.42	39.90
	SSIM	0.9919	0.9918	0.9902
	σ_{sch}	6.10	6.27	1.98
	σ	4.36	4.19	10.31
	Time(s)	0.09	0.004	1.34
Cave	PNSR	47.68	48.20	45.32
	SSIM	0.9964	0.9967	0.9942
	σ_{sch}	4.23	4.73	1.57
	σ	15.20	15.80	21.40
	Time(s)	0.12	0.01	3.40
TokyoTech 31	PNSR	45.78	45.17	44.70
	SSIM	0.9956	0.9950	0.9948
	σ_{sch}	4.29	4.26	4.11
	σ	27.80	20.60	40.06
	Time(s)	0.17	0.01	4.93

this study was for Neural Network training only 6.25% of image database was used. Probably there is potential to do better.

For the RAW images from the 5 channel SFA camera [29], both NN and LMMSE gives artifacts in flat regions of the image. Compared to the algorithm proposed by Monno et al. the output image is sharper with less false colors. Especially if we note the text in the images, its more readable with less false colors around the edges. The artifacts are problematic, however using simple post processing it is possible to remove them. This shows that there is still potential to do better image demosaicing for such SFAs, as information is present in the RAW image. One thing to remember is that the algorithm Weighted Guided Filter proposed by Monno et al. was jointly developed for this SFA arrangement while the LMMSE and NN approach are generic in nature. Also the NN approach is considerably faster compared to WGF therefore making it more suitable for embedded applications. Further LMMSE has an even bigger advantage over Neural Network on execution speed as it is twice as fast. However for demanding applications where image quality is paramount, NN affords more flexibility as by increasing the number of neurons and layers we can further improve the result.

Conclusion

In this paper we presented a dual-layer NN approaching to demosaicing. We presented demosaicing results for RGB, RGBW, 5-channel Color Filter Arrays and a Spectral Filter Array having 5 channels. We showed that our approach gives competitive results compared to state of art for RGB and RGB with panchromatic filters. For Monno5ch SFA prototype camera the output image is sharper, with less false colors but with artifacts. Therefore, we have demonstrated that there is potential to do better as information is present in the RAW image and demosaicing for SFAs is still an interesting problem. The solution proposed by us is a relatively simple NN which is cost/energy efficient com-

pared to other state of art solutions. Also by considering more neurons, adding layers or different topology may give better result which is a work to be pursued in future.

Acknowledgment

This work is financed by the FUI Extrem OWL no. F1411026U to P. Amba and D. Alleysson. I would like to thank Dr. Yusuke Monno of Tokyo Institute of Technology for sharing data regarding their multispectral camera.

References

- [1] Flickr, <https://www.flickr.com>.
- [2] Fuji xtrans, <https://www.fujifilm.eu/uk/products/digital-cameras/model/x-pro1/features-4483/aps-c-16m-x-trans-cmos>.
- [3] Imec hyperspectral imaging, <https://www.imec-int.com/en/hyperspectral-imaging>.
- [4] Kodak image database, <http://r0k.us/graphics/kodak/>.
- [5] Silios color shades, <https://www.silios.com/cms-cameras-1>.
- [6] D. Alleysson, B. De Lavarene, and J. Héroult. Digital image sensor, image capture and reconstruction method and system for implementing same, Oct. 22 2013. US Patent 8,564,699.
- [7] D. Alleysson, S. Susstrunk, and J. Héroult. Linear demosaicing inspired by the human visual system. *IEEE Transactions on Image Processing*, 14(4):439–449, 2005.
- [8] P. Amba, J. Dias, and D. Alleysson. Random color filter arrays are better than regular ones. *Journal of Imaging Science and Technology*, 60(5):50406–1, 2016.
- [9] P. Amba, J. B. Thomas, and D. Alleysson. N-lmmse demosaicing for spectral filter arrays. *Journal of Imaging Science and Technology*, 61(4):40407–1, 2017.
- [10] B. Bayer. Color imaging array, July 20 1976. US Patent 3,971,065.
- [11] B. C. De Lavarène, D. Alleysson, and J. Héroult. Practical implementation of lmmse demosaicing using luminance and chrominance spaces. *Computer Vision and Image Understanding*, 107(1):3–13, 2007.
- [12] G. D. Finlayson, S. D. Hordley, and P. Morovic. Using the spectracube to build a multispectral image database. In *Conference on Colour in Graphics, Imaging, and Vision*, volume 2004, pages 268–274. Society for Imaging Science and Technology, 2004.
- [13] M. Gharbi, G. Chaurasia, S. Paris, and F. Durand. Deep joint demosaicking and denoising. *ACM Transactions on Graphics (TOG)*, 35(6):191, 2016.
- [14] B. K. Gunturk, Y. Altunbasak, and R. M. Mersereau. Color plane interpolation using alternating projections. *IEEE transactions on image processing*, 11(9):997–1013, 2002.
- [15] K. Hirakawa and T. W. Parks. Adaptive homogeneity-directed demosaicing algorithm. *IEEE Transactions on Image Processing*, 14(3):360–369, 2005.
- [16] I. Hirota. Solid-state imaging device, method for processing signal of solid-state imaging device, and imaging apparatus, May 7 2013. US Patent 8,436,925.
- [17] S. Hordley, G. Finalyson, and P. Morovic. A multi-spectral image database and its application to image rendering across illumination. In *Image and Graphics (ICIG'04), Third International Conference on*, pages 394–397. IEEE, 2004.
- [18] A. Hore and D. Ziou. An edge-sensing generic demosaicing algorithm with application to image resampling. *IEEE Transactions on Image Processing*, 20(11):3136–3150, 2011.
- [19] I. Kanellopoulos and G. Wilkinson. Strategies and best practice for

- neural network image classification. *International Journal of Remote Sensing*, 18(4):711–725, 1997.
- [20] T. Kijima, H. Nakamura, J. Compton, J. Hamilton, and T. DeWeese. Image sensor with improved light sensitivity, Nov. 22 2007. US Patent App. 11/419,574.
- [21] X. Li. Demosaicing by successive approximation. *IEEE Transactions on Image Processing*, 14(3):370–379, 2005.
- [22] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 2272–2279. IEEE, 2009.
- [23] H. S. Malvar, L.-w. He, and R. Cutler. 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, pages iii–485. IEEE, 2004.
- [24] D. Menon, S. Andriani, and G. Calvagno. Demosaicing with directional filtering and a posteriori decision. *IEEE Transactions on Image Processing*, 16(1):132–141, 2007.
- [25] L. Miao, H. Qi, and R. Ramanath. A generic binary tree-based progressive demosaicking method for multispectral filter array. In *International Conference on Image Processing (ICIP), 2006*, pages 3221–3224. IEEE, Oct 2006.
- [26] L. Miao, H. Qi, and R. Ramanath. Generic msfa mosaicking and demosaicking for multispectral cameras. In *Digital Photography II*, volume 6069 of *Proc. SPIE*, pages 606909–606909–10, 2006.
- [27] L. Miao, H. Qi, R. Ramanath, and W. E. Snyder. Binary tree-based generic demosaicking algorithm for multispectral filter arrays. *IEEE Transactions on Image Processing*, 15(11):3550–3558, Nov 2006.
- [28] Y. Monno, D. Kiku, S. Kikuchi, M. Tanaka, and M. Okutomi. Multispectral demosaicking with novel guide image generation and residual interpolation. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 645–649. IEEE, 2014.
- [29] Y. Monno, S. Kikuchi, M. Tanaka, and M. Okutomi. A practical one-shot multispectral imaging system using a single image sensor. *IEEE Transactions on Image Processing*, 24(10):3048–3059, Oct 2015.
- [30] Y. Monno, M. Tanaka, and M. Okutomi. Multispectral demosaicking using adaptive kernel upsampling. In *18th IEEE International Conference on Image Processing (ICIP)*, pages 3157–3160, Brussels, Sept. 2011.
- [31] Y. Monno, M. Tanaka, and M. Okutomi. Multispectral demosaicking using guided filter. In S. Battiato, B. G. Rodricks, N. Sampat, F. H. Imai, and F. Xiao, editors, *Digital Photography VIII*, volume 8299 of *Proc. of SPIE*, pages 829900–829900–7, Burlingame, California, USA, Jan. 2012.
- [32] D. Paliy, A. Foi, R. Bilcu, and V. Katkovnik. Denoising and interpolation of noisy bayer data with adaptive cross-color filters. In *Electronic Imaging 2008*, pages 68221K–68221K. International Society for Optics and Photonics, 2008.
- [33] T. Skauli and J. Farrell. A collection of hyperspectral images for imaging systems research. volume 8660 of *Proc. SPIE*, pages 86600C–86600C–7, 2013.
- [34] D. Taubman. Generalized wiener reconstruction of images from colour sensor data using a scale invariant prior. In *Image Processing, 2000. Proceedings. 2000 International Conference on*, volume 3, pages 801–804. IEEE, 2000.
- [35] J.-B. Thomas, P.-J. Lapray, P. Gouton, and C. Clerc. Spectral characterization of a prototype SFA camera for joint visible and nir acquisition. *Sensors*, 16(7):993, 2016.
- [36] L. Wang and G. Jeon. Bayer pattern cfa demosaicking based on multi-directional weighted interpolation and guided filter. *IEEE Signal Processing Letters*, 22(11):2083–2087, 2015.
- [37] Y.-Q. Wang. A multilayer neural network for image demosaicking. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 1852–1856. IEEE, 2014.
- [38] F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar. Generalized assorted pixel camera: Post-capture control of resolution, dynamic range and spectrum. Technical report, Department of Computer Science, Columbia University CUCS-061-08, Nov 2008. <http://www1.cs.columbia.edu/CAVE/databases/multispectral/>.
- [39] F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar. Generalized assorted pixel camera: postcapture control of resolution, dynamic range, and spectrum. *IEEE transactions on image processing*, 19(9):2241–2253, 2010.
- [40] C. Zhang, Y. Li, J. Wang, and P. Hao. Universal demosaicking of color filter arrays. *IEEE Transactions on Image Processing*, 25(11):5173–5186, 2016.
- [41] L. Zhang and X. Wu. Color demosaicking via directional linear minimum mean square-error estimation. *IEEE Transactions on Image Processing*, 14(12):2167–2178, 2005.
- [42] L. Zhang, X. Wu, A. Buades, and X. Li. Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. *Journal of Electronic imaging*, 20(2):023016–023016, 2011.
- [43] R. Zhou, R. Achanta, and S. Süsstrunk. Deep residual network for joint demosaicking and super-resolution. *arXiv preprint arXiv:1802.06573*, 2018.

Author Biography

Prakhar Amba received his B.Tech from DA-IICT Gandhinagar, India (2007), his M.Sc from ISEP, Paris, France (2009) and his Phd. from Université Grenoble Alpes, France (2018). His research interest includes color vision and image processing.

David Alleysson received his B.Tech in computer science from Ecole d'ingénieurs de Genève in 1993, his Msc. from Ensimag in 1994 and his PhD from Université Joseph Fourier in 1999, working with Jeanny Hérault in color vision. He then spend three years in EPFL Lausanne, working on color printing and digital cinema and color vision with Sabine Susstrunk. From 2003 he is CNRS researcher at LPNC. His research interest is the modeling of color vision for improving digital color image processing, demosaicing and gamut mapping.

Martial Mermillod's (Pr. Univ. Grenoble Alpes, CNRS, LPNC & LJK, Grenoble, France) research focuses on neural computation, cognitive neuroscience and psychology, and applied to the field of cognitive and perceptual processes. He has published more than 60 articles in high-ranked journals and several proceedings and books.