CFA Bayer image sequence denoising and demosaicking chain

A. Buades  J. Duran

Abstract

The demosaicking provokes the spatial and color correlation of noise, which is afterwards enhanced by the imaging pipeline. The correct removal previous or simultaneously with the demosaicking process is not usually considered in the literature. We present a novel imaging chain including a denoising of the Bayer CFA and a demosaicking method for image sequences. The proposed algorithm uses a spatio-temporal patch method for the noise removal and demosaicking of the CFA. The experimentation, including real examples, illustrates the superior performance of the proposed chain, avoiding the creation of artifacts and colored spots in the final image.

1 Introduction

Digital photographs are usually represented by three color values at each pixel. However, most common cameras use a CCD or CMOS sensor device measuring a single color per pixel. Demosaicking is the interpolation process by which the two missing color values are estimated. The selected configuration of the sensor usually follows the Bayer color filter array (CFA) [5]. Thus, out of a group of four pixels, two are green (in quincunx), one is red and one is blue [1].

The demosaicking is usually performed by combining close values from the same channel or the other two. As a result, the noise, being almost white at the sensor, gets correlated. The rest of the imaging chain, consisting mainly in color and gamma corrections and compression, enhances the noise in dark parts of the image leading to contrasted colored spots of several pixels. The size of these spots depends on the applied demosaicking method. The removal of these artifacts after the imaging chain is applied is a difficult task, since structure and noise are not easily differentiated. As a consequence, details and texture might be attenuated during the denoising process, when having access only to this processed data.

In dark and indoor scenes, the limitations of imaging pipelines are more noticeable. If the camera is set to a long exposure time, the photograph gets blurred by the camera motion and aperture. If it is taken with short exposure, the image is dark, and enhancing it reveals the noise. The use of a high ISO value for short exposures makes the image brighter but is makes noise more annoying. Many times, the dilemma can be solved by taking a burst of images. A “burst”, or “image burst” means a set of digital images taken from the same camera, in the same state, and quasi instantaneously. Such bursts are obtained by video, or by using the burst mode proposed in recent reflex and compact cameras, or simply by taking several snapshots while holding firmly the camera in a fixed direction.

We propose a multi-image processing chain, consisting of a novel denoising method for the removal of noise at the CFA sensor data and a novel multi-image demosaicking algorithm. The denoising method deals automatically with the incomplete CFA and unknown signal dependent noise distribution. It is based on the white noise removal algorithm for video [10]. This algorithm groups similar spatio-temporal patches and removes noise by thresholding in an adapted PCA basis.
The novel demoisaicking method is also patch based. It combines similar patches by averaging, introducing the CFA Bayer pattern as a restriction to ensure that only initial values are combined. The demosaicking process interpolates missing values but also modifies CFA ones. This permits the removal of residual noise and avoids the creation of zipper effect. The zipper effect is an on-off pattern created by the juxtaposition of contiguous original and interpolated pixels. The proposed process is applied first to the green channel of the sequence, and then to the difference between red/blue channels and the updated green.

This paper is organized as follows. In Section 2 we present the literature on image sequence and video denoising. The new algorithm for denoising a sequence of color filter array (CFA) images is introduced in Section 3. Section 4 introduce the novel image sequence demoisaicking method. Section 5 shows the performance of the proposed method by means of a comparison with state of the art approaches. Finally, Section 6 concludes the paper.

2 State of the Art

2.1 Image demosaicking

Color demosaicking techniques have been extensively reviewed in the literature [46, 38]. Due to the CFA configuration, the interpolation of the green channel is commonly performed in a first step. Then, in order to take advantage of inter-channel correlation, the differences or ratios between the red/blue and the estimated green are interpolated instead of the red/blue channel directly. Former demosaicking methods [27] concentrated on locally estimating the most suitable direction of interpolation for the green channel. In many cases, the ambiguity of the local configuration in the CFA makes nearly impossible to decide properly. Many authors [61, 45] proposed to take this decision a posteriori once the green channel or even the full color image has been interpolated completely horizontally and completely vertically.

Local demosaicking strategies can create artifacts such as aliasing, erroneously interpolated structures or zipper effect. Buades et al. [8] showed that these interpolation artifacts can be eliminated by involving image self-similarity and redundancy. In a posterior paper [19], it was shown that nonlocal filtering achieves better results if applied to channel differences instead of channels themselves.

2.2 Image sequence demosaicking

Despite the extensive literature in single color image demosaicking, there exist few works on its extension to video. While the demosaicking of a single image might give reasonable results, the consecutive play of several frames might introduce visual artifacts. Wu et al. [57] match the CFA green sample blocks in adjacent frames via motion analysis and fuse them with intra-frame estimates of the missing green samples. Lukac et al. [42] introduced a video demosaicking method that uses a set of stencils comprising three consecutive frames. Gevrekci et al. [24] proposed an image sequence extension of the projection onto convex sets algorithm [26] by adding a new constraint set based on the spatio-intensity neighborhood. Vandewalle et al. [55] aligns the set of image looking for a rotation plus translation in the fourier domain. After image registration, a demosaicked image is reconstructed at once using the full set of images. For this, the authors use normalized convolution, an image interpolation method from a nonuniform set of samples.

Although it is beyond the scope of this paper, let us mention that video demosaicking is often proposed conjointly with super-resolution [25, 22, 54]. Gotoh et al. [24] introduced a variational method accounting for motion, subsampling and mosaicking with edge-adaptive regularization for luminance component and isotropic regularization for chrominance components. Farsiu et al. [22] proposed a hybrid method for super-resolution
and demosaicking based, with the limitation of the motion field to be translational, on a MAP estimation technique by minimizing a cost function with bilateral regularization on the luminance component and Tikhonov regularization on the chrominance ones. An extra regularization term is incorporated to force similar edge location and orientation in different color channels. Karch et al. [34] first demosaick all frames individually and interpolate all of them to the desired resolution. Then, a weighted sum of the interpolated frames is used to fuse them into an improved resolution estimate. Finally, restoration is applied to improve any degrading camera effects.

2.3 Image sequence white noise removal

Local average methods, as the bilateral filter [54], or patch based methods as NL-means [9] or BM3D [16] and NLBayes [30] can be easily adapted to video just by extending the neighboring area to the adjacent frames. Kervrann and Boulanger [7] extended the NL-means to video by growing adaptively the spatio-temporal neighborhood. Arias et al. extended the NL-Bayes [30] to video [8, 9]. Methods based in sparse decompositions are extended to image sequences [41, 53, 56, 37], as well as approaches based on low rank approximation [31, 32].

Other methods combine a single image estimate with a purely temporal one. Dai et al. [18] applied a purely temporal LMMSE estimate on motion trajectories. Yue et al. [59] used a BM3D estimate of each frame and a BM3D applied to similar patches of neighboring frames. Dai et al. [17] adds intercolor prediction to previous approaches.

The performance of many denoising methods is improved by introducing motion compensation. These compensated filters estimate explicitly the motion of the sequence and compensate the neighborhoods yielding stationary data [48]. The BM3D extension, VBM4D [43], exploits the mutual similarity between 3-D spatio-temporal volumes constructed by tracking blocks along trajectories defined by the motion vectors. In [10] the authors proposed to combine optical flow estimation and patch based methods for denoising. Their algorithm tends to a fusion of the neighboring frames in the absence of occlusions and a dense temporal sampling. In this ideal scenario, an optical flow or global registration is able to align the frames and fusion is achieved by simple averaging [28]. The algorithm in [10] compensates the failure of these requirements by introducing spatiotemporal patch comparison and denoising in an adapted PCA based transform.

2.4 Image sequence color and spatially correlated noise removal

It must be noted that the previous techniques apply only to additive uniform white noise, and not to real photography and video data. The literature on noise removal from video is scarce. Bennet and McMillan [6] apply a spatiotemporal bilateral filtering combined with Durand et al. [20] enhancement technique. Filtering parameters at each pixel are fixed depending on the amount of enhancement to be applied. Liu et al. [30] proposed a strategy using motion compensation and the NL-means algorithm. Xu et al. [58] apply the NL-means denoising algorithm to each image and then combine this estimate with a purely temporal application of the same algorithm. Both estimates are combined in static parts of the image, while the spatial estimate is preferred in moving ones. A second iteration of the denoising algorithm is applied after a contrast enhancement stage. A similar approach is used by Kim et al. [35] but a purely temporal denoising algorithm is applied first, and a second purely spatial after the enhancement stage. Gao et al. [23] perform a bilateral filter in both luminance and chrominance (YCbCr) for each frame, for which the bilateral weight distribution is computed only on the luminance. A multiscale wavelet transform permits to deal with non white noise. Jovanov et al. [33] simultaneously perform multiview image sequence denoising, color correction and the improvement of sharpness in slightly defocused regions for sequences of images coming from different cameras and thus with different
degrees of blur and noise. In [11] the authors proposed a multi-scale algorithm, estimating and removing noise at each scale. It uses a variance stabilization transform and the white noise removal algorithm in [10].

A particular type of algorithms are those dealing with a burst of images. That is a sequence of images taken nearly instantaneously with close optical centers. For such type of images, a parametric transform, commonly an homography, is able to register the images of the sequence. A robust combination of these images after registration permits to remove correlated noise [28]. A fast version by Liu et al. [40] accelerates the homography selection by a multi scale strategy. The multiscale procedure includes a local refining of the homography and the image combination.

2.5 Noise removal at the CFA and joint denoising-demosaicking

Since demosaicking is the main cause of noise correlation, it is suitable to remove the noise before this process, or simply combine them in a single procedure.

Paliy et al. [49] performs interpolation and denoising of a single CFA Bayer image. The method uses inter color filters selected by polynomial approximation. Chatterjee et al. [13] denoises the CFA by using the NL-means algorithm. Their method averages only patches having the same CFA pattern. The authors do not use any variance stabilization transform. They propose a demosaicking method based on an optimization process where the CFA is taken as the low resolution counterpart in a super resolution framework.

Zhang et al. [62] proposed to first denoise the CFA before applying a spatio-temporal demosaicking algorithm. A spatio temporal extension of [63] is applied for denoising by combining only patches having the same CFA pattern. Noise is reduced by thresholding in an adaptive PCA basis. Since each patch belongs to the CFA actually contains red, green and blue pixels. For demosaicking, an initial demosaicking is processed by a spatial-temporal algorithm in order to reduce artifacts. For a given pixel to be processed, one searches for the similar pixels to it within the spatial-temporal neighborhood and then let the enhanced pixel be the weighted average of them.

Heide et al. [30] proposed a single optimization process being able to deal with all image chain stages. Denoising, demosaicking and deconvolution are written in a single energy minimization solved by primal-dual techniques [12]. The method applies to a single image, even it can be adapted to deal with image sequences. Patil et al. [50] denoise the CFA sensor data by a dictionary learning combined with a variance stabilization transform. No particular demosaicking algorithm is proposed. Hasinoff et al. [29] proposed a chain for burst sequences of images, with the novelty of dealing with the Bayer data. The method aligns and merges all images of the burst sequence into a single one. Subsampled Bayer of factor 2 is used for alignment, a tiled translation is estimated by a Gaussian pyramid process. Finally, a robust merging method is implemented using the FFT.

Neural networks have also been recently proposed for burst fusion Mildenhall et al. [47].

3 CFA Denoising

We propose to denoise the sequence of CFA images before the demosaicking. In order to do so, instead of dealing directly with the CFA structure, we subsample each image into a 4-channel one, accounting for the red, two greens and blue values. Each new image has half the width and height of the original sensor data. Now, a sequence of complete 4-channel images has to be denoised.
3.1 Noise estimation

In order to estimate noise, we assume that all images of the sequence have been acquired with the same conditions, that is, the same ISO gain factor and exposure. In such a case, all images of the sequence share the same noise model.

We adapt the single image noise estimation algorithm in [14]. This algorithm divides each image in patches and applies the DCT as proposed by Ponomarenko [51] for uniform noise estimation. The low frequencies of the DCT permit to select the less oscillating patches, and the high frequencies of these selected patches yield a standard deviation estimate.

We classify patches from all the sequence depending on its mean. The full color range is partitioned into non overlapping bins, and each patch is arranged into the corresponding one. This permits to estimate a standard deviation for each bin, and thus an intensity dependent model. Since noise at the sensor is white, this algorithm yields a correct estimate. For spatially correlated noise, algorithms exploiting temporal standard deviation should be applied [11]. Since noise is independent for different images, such algorithms provide a more accurate estimate for correlated noise. In our case, we prefer the first strategy since does not involve any registration stage, making the process faster. The algorithm is applied independently to each of the four channels.
The algorithm yields a set of observations \( \{ x_i, \sigma_i \} \) which has to be interpolated to the whole range in order to have a complete noise model. Noise at the sensor is often assumed to follow a Poisson distribution, with linear variance. However, this simplification is not valid for dark values where other sources of noise are dominant. In Figure 1, we illustrate how the obtained set \( \{ x_i, \sigma_i \} \) is not well fitted by a single linear model. Instead, we preferred to use two different linear models, one for the dark and another one for lighter parts of the color range. We use a minimum least square method for optimizing such a model. We ask the two linear models to be jointly continuous, and the point in the range for which the approximation changes is also considered as a part of the optimization problem. Figure 1 compares the single linear approach to the proposed one. It is observed how this interpolation permits to correctly fit the noise model in the whole range. A quadratic model could be used for fitting the darker part, but in such case we should require the coefficient of degree 2 to be negative. We kept the simplest piecewise linear model.

### 3.2 Variance stabilization

Since noise at the CFA is signal dependent we may apply a variance stabilization transform. We usually refer to the Anscombe transform as the transformation \( f(u) = 2\sqrt{u + \frac{3}{8}} \) which is known to stabilize the variance of a Poisson noise model. However, any signal dependent additive noise can be stabilized by a simple transform. Let \( v = u + g(u)n \) be the noisy signal, we search for a function \( f \) such that \( f(v) \) has uniform standard deviation. When the noise is small compared to the signal we can apply the decomposition \( f(v) = f(u) + f'(u)g(u)n \). Forcing the noise term to be constant, \( f'(u)g(u) = c \), and integrating we obtain

\[
f(u) = \int_0^u \frac{c dt}{g(t)}.
\]

When a linear variance noise model is taken, this transformation gives back the known Anscombe transform. The inverse transform is applied back after denoising to get the original range.

In Figure 2 we compare the application of the classical Anscombe transform with the proposed variance stabilization. In order to do so, we estimate the noise for a single Nikon CFA image with the proposed approach. In order to apply the proposed variance stabilization, we set the parameter \( c \), the amplitude of the noise after stabilization in such a way the color range of the transformed image equals the range of the stabilized with the classical transform. In this way, we can now estimate the noise in these transformed images. We use the signal dependent noise estimation [14] with 5 bins, and display this estimation in the same figure. We appreciate, how noise amplitude is constant for the whole range after the proposed stabilization, which is not the case of the classical one.

Since noise amplitude is different, we apply for each channel its own estimation, which is not the case of the classical anscombe which applies the same transformation for all channels.

### 3.3 Noise removal

We will adapt the image sequence denoising method in [10]. We summarize the complete algorithm for denoising a frame \( I_k \) from a sequence \( \{ I_1, I_2, \cdots, I_N \} \). The same procedure is applied sequentially to all the frames of the sequence.

First, the optical flow between \( I_k \) and adjacent frames in a temporal neighborhood is computed and used for warping these frames onto \( I_k \). Occlusions are detected depending on the divergence of the computed flow: negative divergence values indicate occlusions. Additionally, the color difference is checked after flow compensation. A large difference indicates occlusion, or at least failure of the color constancy assumption.
Figure 2: Anscombe analysis. We apply the classical anscombe and proposed variance stabilization transform to a CFA image. Image is first sampled to be a full color image. Left: image after classical anscombe. Middle: image after proposed stabilization. Right: noise estimation in stabilized images. The dashed line (proposed estimation) is clearly more constant the continous line (classical anscombe).

Once the neighboring frames have been warped, the algorithm uses a 3D volumetric approach to search for similar patches, while still 2D image patches are used for denoising. For each patch $P$ of the reference frame $I_k$, the patch $\mathcal{P}$ referring to its extension to the temporal dimension is considered, having $M$ times more pixels than the original one (assuming $M$ patches in the temporal neighborhood). Since the images have been resampled according to the estimated flow, the data is supposed to be static. The algorithm looks for the $K$ extended patches closest to $\mathcal{P}$. As each extended patch contains $M$ 2D image patches, the group contains $K \cdot M$ selected patches. The Principal Component Analysis (PCA) of these patches is computed and their denoised counterparts are obtained by thresholding of the coefficients. As proposed in [63], the decision of canceling a coefficient is not taken depending on its magnitude, but the magnitude of the associated principal value. A more robust thresholding is obtained by comparing the principal values to the noise standard deviation and canceling or maintaining the coefficients of all the patches associated to a certain principal direction. The whole patch is restored in order to obtain the final estimate by aggregation.

Color denoising is achieved by considering color-patches with three times more samples than in the gray case. This permits to adapt both to geometry and color correlation among selected patches. However, this strategy has the disadvantage of being quite slow. In our case, for a set of patches of $n \times n$ pixels, this involves the computation of the PCA for vectors of size $4n^2$. In addition in order to correctly compute such a representation we need the set of patches to be larger than the dimension of the vectors. Both factors makes the algorithm slow for the removal of noise for a sequence of nowadays camera resolutions. The method in [10] also involves a second oracle step, in which the first denoised sequence is used to drive a second iteration of the method. We drop this second iteration in order to reduce the time of computation.

We will use a decorrelation transform YUVW for each 4-channel image. In order to estimate this transform, we took several CFA images, converted to 4-channel ones and consider for each pixel the four dimensional vector containing its red, two green and blue values. The PCA analysis of this set yields the transform

$$M = \begin{pmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ -0.5 & 0.5 & 0.5 & -0.5 \\ 0.65 & 0.2784 & -0.2784 & -0.65 \\ -0.2784 & 0.65 & -0.65 & 0.2784 \end{pmatrix},$$

where each row of the matrix represents a principal vector. Similar analysis to standard color images yields an
orthonormal version of the usual YUV color space. Since this is matrix an orthonormal matrix, its application after the variance stabilization transform makes the new components to have uniform noise with the same standard deviation. The transposed matrix yields the inverse transform.

For each new channel YUVW, the sequence is denoised separately. However, the component $Y$ is used for registering the frames and to guide patch selection for all channels. After all components have been denoised, the inverse decorrelation transform is applied, as well as, the inverse variance stabilization transform to each channel.

4 Demosaicking

We introduce an image sequence algorithm to both interpolate missing values and remove residual noise from all pixels. The algorithm makes use of optical flow and exploits spatio-temporal redundancy with patch-based techniques to correct an initial interpolation. A decimation mask $D$ keeps the trace of initial CFA values and permits to differentiate from the initially interpolated ones.

We apply this algorithm to generate a demosaicked and denoised video sequence from $\{f_n = (f_n^R, f_n^G, f_n^B)\}_{n=1}^N$ the mosaicked and eventually noisy frames. The same strategy could be adapted for other filtering/interpolation tasks as video deinterlacing or the increase of temporal resolution.

4.1 Spatio-temporal interpolation

Below we detail the algorithm to remove noise and aliasing from the initially interpolated sequence, denoted by $\{\tilde{f}_n\}_{n=1}^N$. This sequence might be obtained by bicubic interpolation, any anisotropic or demosaicking method. The algorithm describes the processing of a reference frame $\tilde{f}$. Even if this description applies also to multivalued data, we will apply it to single channel images.

In order to process $\tilde{f}$, we need to establish a correspondence function between this frame and all the other ones in the sequence. We use the TV-L1 optical flow introduced in [60].

The goal is to remove the aliasing and increase the quality of the initially interpolated images $\tilde{f}_1, \ldots, \tilde{f}_N$ by weighted averaging. The algorithm proceeds patch per patch of the reference interpolated frame $\tilde{f}(P)$ by computing

$$\tilde{u}(P) = \frac{1}{C} \sum_{P_n \in N_P} w(\tilde{f}(P), \tilde{f}_n(P_n^k)) \, D(P_n^k) \cdot \tilde{f}_n(P_n^k),$$

being $D$ the decimation mask associated to the sampling operator and $w(\tilde{f}(P), \tilde{f}_n(P_n^k))$ a real number measuring the similarity between patches $\tilde{f}(P)$ and $\tilde{f}_n(P_n^k)$. The number $n$ denotes the index of the image which $P_n^k$ belongs
to. In this setting, \( C \) is a normalization factor and the operator \( \cdot \) denotes the product element by element of each patch. The division by the normalization patch \( C \) is also made element by element:

\[
C = \sum_{P_k^n \in \mathcal{N}_P} w(\hat{f}(P), \hat{f}_n(P_k^n)) D(P_k^n).
\] (2)

The use of the decimation mask \( D \), which is assumed to be the same for the sake of simplicity (the algorithm performs in a similar manner if the downsampling operator differs for each frame), makes the algorithm average only initial values.

The selection of candidate patches in \( \mathcal{N}_P \) actually depends on a 3D distance taking into account motion estimation. This makes the selection procedure more robust to noise and aliasing artifacts. For each reference patch \( \hat{f}(P) \), we denote as \( \mathcal{P} \) its motion-compensated extension to the temporal dimension, having \( N \) times more pixels than the original one:

\[
\mathcal{P} = \bigcup_{n=1}^{N} (P + W_n^\top(P)),
\]

where \( W_n^\top \) is a motion shift for patch \( P \) and \( n \)-th frame. In practice, we take this shift to be the estimated forward flow between images \( \hat{f} \) and \( \hat{f}_n \) at the pixel in the centre of patch \( P \).

The algorithm looks for the \( K \) extended patches closest to \( \mathcal{P} \) minimizing the distance

\[
d(\mathcal{P}, \mathcal{Q}) = \sum_{n=1}^{N} \| \hat{f}_n(P + W_n^\top(P)) - \hat{f}_n(Q + W_n^\top(Q)) \|^2.
\] (3)

As each extended patch contains \( N \) 2D patches, the selected group \( \mathcal{N}_P \) contains \( K \cdot N \) patches,

\[
\{ P_k^n = P_k + W_n^\top(P_k) \mid n = 1, \ldots, N, k = 1, \ldots, K \}.
\]

We compute the similarity of each of these 2D patches and \( P \) as

\[
w\left( \hat{f}(P), \hat{f}_n(P_k^n) \right) = \exp \left( -\frac{\| \hat{f}(P) - \hat{f}_n(P_k^n) \|^2}{h^2} \right).
\] (4)

The value of \( h \) depends on the degree of aliasing and the noise statistics. The preselection procedure using motion compensation makes this value less critical than in other patch-based regularization techniques \cite{21, 52}.

Finally, each pixel is estimated by aggregating all the values obtained by all patches containing it.

Importantly, no occlusion detection is performed on the estimated flow. In addition, as occlusion regions might be different from one frame to the other, it makes no sense to use the distance \( d(\mathcal{P}, \mathcal{Q}) \) for computing the final weight. The comparison between patches \( \hat{f}(P) \) and \( \hat{f}_n(P_k^n) \) acts as a validation stage and avoids averaging very different patches.

### 4.2 Initializations and flow estimation

We first compute initial demosaicked images \( \{ \hat{f}_n \}_{n=1}^{N} \). For each frame, we use the local directional interpolation method proposed in \cite[Section II]{19}. We apply this demosaicking algorithm to the output of our CFA denoising algorithm. The initially demosaicked image may contain typical demosaicking artifacts and eventually residual noise.
The method in [19, Section II] first estimates the green channel. At each red and blue position determined by the CFA, four interpolated values of the green are computed along north, south, east and west directions. Instead of deciding for each pixel which is the dominant direction and interpolate according to it, this decision is taken a posteriori once a full color image has been reconstructed for each direction. Therefore, for each of these directionally interpolated green channels, the red and blue components are reconstructed by bilinear interpolation on the channel differences green-red and green-blue. Finally, a decision of the most suitable approximation is made according to the variation of the chrominance (measured in the YUV space) at each pixel along the four directions.

Once the initial demosaicked video has been generated, we proceed to compute the optical flow between each pair of images in the sequence. Due to the higher sampling rate of the green component which permits an easier reconstruction of the main geometry and texture than the red and blue channels, the optical flow is computed on the sequence of interpolated green images \( \{ \tilde{f}_G^N \}_{n=1}^N \).

### 4.3 Spatio-temporal demosaicking

The green channel of each frame is first updated following the described spatio-temporal interpolation method applied to \( \{ \tilde{f}_G^N \}_{n=1}^N \). The mask \( D \) used in (1)-(2) is the CFA mask corresponding to the green channel, i.e., a quinqux of factor 2 for each line and column. Therefore, only original green values are used in the weighted averaging. Furthermore, the patch-based Euclidean distances are computed on the initially interpolated green sequence \( \{ \tilde{f}_G^N \}_{n=1}^N \).

Once all green channels have been updated, we apply the same strategy independently to the red and blue channels. Following [19], we apply a non local average to the channel differences red-green and blue-green instead of the red and blue themselves. For the sake of simplicity, we shall describe only the process for the red channel.

We consider the red channels of the initially interpolated frames \( \{ \tilde{f}_R^N \}_{n=1}^N \) and we compute at each pixel the difference with the updated green, that is, \( \{ f_n - u_G^N \}_{n=1}^N \). We apply now the spatio-temporal interpolation method by considering \( \{ f_n - u_G^N \}_{n=1}^N \) as the input grayscale sequence. The CFA mask of the red channel is used as mask \( D \). Furthermore, the patch-based Euclidean distances in (3)-(4) are computed on the updated green sequence \( \{ u_G^N \}_{n=1}^N \) instead of the channel difference. Once the method has been applied and the channel differences have been updated at each pixel by patch aggregation, the green value is added back to get the final red component. This process is performed on each frame, so we get the final red channels \( \{ u_R^N \}_{n=1}^N \).

The application of the method to the green channels removes any residual noise kept at the CFA denoising, as it does the posterior application to the channel differences of the red and blue with the updated green.

### 5 Discussion and Experimental Results

We test the performance of our method for both simulated and real data.

#### 5.1 Simulated data

We simulate noisy CFA sequences from a dataset of full color ones. All sequences are composed of 8 frames, the central frame of each one is displayed in Figure 3. We subsample the images according to the Bayer pattern and add noise of different standard deviations. In this case, as images are already in color, we do not apply any particular imaging chain, thus colored spots are not enhanced by the white balance or gamma correction. Since we have the ground truth we evaluate the results both visually and numerically.
Figure 4: Visual comparison with simulated noisy CFAs. From left to right: image by local demosaicking, image by temporal demosaicking plus VBM3D, proposed demosaicking, proposed CFA denoising plus local demosaicking and complete chain. Below: detail of above images.

We compare the proposed chain with,

- The local interpolation method introduced in [19, Section II], which is used as initialization for the demosaicking stage without any denoising step.

- The combination of the temporal demosaicking method introduced in [57] with a posterior denoising using VBM3D [15]. Since the standard deviation of noise is modified after the demosaicking process, we tested several parameters for VBM3D and kept the one with smallest error. The VBM3D is a video restoration method grouping patches of consecutive frames and performing transform thresholding. It also has a second oracle iteration using a first estimate to drive the patch selection and thresholding. Our algorithm does not involve any second iteration in order to reduce computation time.

- The application of the proposed demosaicking strategy without any previous denoising. Since the demosaicking method modifies all values, CFA values inclusive, it actually removes noise. In such a case, the parameter $h$ is fixed depending on the noise standard deviation.

- The proposed CFA denoising with the posterior local interpolation [19, Section II].

- The full proposed chain. In this case, the demosaicking algorithm only filters residual noise not completely removed by the denoising stage.

Table 1 displays the root mean square error (RMSE) for the central view of the sequence and noise standard deviations $\sigma = 5$ and 10. The error of the local initialization is significantly larger than the other methods since it does not contain any denoising stage.

For low values of $\sigma$, the proposed demosaicking method without denoising, improves the temporal demosaicking plus BM3D and the application of the proposed CFA denoising with the local demosaicking method. This illustrates the performance of the demosaicking method, and its robustness to noise. For $\sigma = 10$ the proposed demosaicking still performs better than the temporal demosaicking plus VBM3D. However, the proposed CFA denoising with the single image local demosaicking outperforms the proposed demosaicking, as well as, the temporal demosaicking plus VBM3D. For both noise standard deviations, the proposed chain combining denoising and demosaicking largely outperforms the rest of algorithms.
Figures 4 and 5 display the results of each method for two sequences. The local demosaicking does not remove any noise. The images processed by the temporal demosaicking and VBM3D are slightly blurry and many details have been removed. Since noise is modified by the demosaicking schemes, its removal is quite challenging. The proposed demosaicking alone is able to correctly remove typical demosaicking artifacts, but is slightly noisy. The proposed chain is able to correctly demosaick and remove noise.

5.2 Real data

We test the proposed method with real data. We use several image sequences acquired with a reflex camera and several RAW video sequences introduced in [2]. Since we need to have access to the RAW data, we could not acquire video ourselves but only images acquired consecutively with a Nikon D80. Some of these data are actually burst sequences, meaning images are acquired quasi instantaneously holding the camera in the same position. Since the camera is hand held, the view point and orientation of the camera slightly changes. For these examples a standard imaging chain composed of white balance by gray mean, and a gamma correction with $\gamma = 0.5$ are used. We do not apply any color correction, in order to convert the RGB values of the sensor to the one of the display.

For real video, we use the video sequences in RAW format introduced in [2]. The imaging pipeline applied after the demosaicking, for these videos, is also made available in the same publication. The sequences used in this section are displayed in Figure 6. Demosaicking and the color chain have been applied for visualization purposes, although the RAW data is the one actually being used.

We compare the application of the standard imaging chain with local demosaicking, the proposed complete chain and the method in [11]. The method in [11] removes noise after the imaging pipeline is applied. The method performs in a multi-scale framework, where noise is estimated at each scale, and the video denoising method in [10] is used after variance stabilization.

Figure 7 compares the application of these methods to a sequence of images acquired with the Nikon D80. Despite being an indoor sequence, the scene is quite illuminated and the noise is moderate. The colored noise spots of the non denoised image are easily noticeable. The denoised sequence by using [11], having access only to the images after the imaging pipeline is applied, is not able to completely remove noise, and isolated color spots are still present.
Table 1: Table 1 displays the root mean square error (RMSE) for the central view of the sequence and noise standard deviations $\sigma = 5$ and 10.

Table 1: Table 1 displays the root mean square error (RMSE) for the central view of the sequence and noise standard deviations $\sigma = 5$ and 10.

spots remain. The proposed chain is able to completely remove noise while keeping the main details.

Figure 5 displays a similar experience but with darker conditions. The image with standard imaging chain illustrates the poor signal to noise ratio in these conditions, for which image details are hardly visible. The proposed algorithm is able to completely remove the noise, making many details appear despite they were hidden by noise. The method in [11] is not able to correctly remove noise.

5.3 Burst sequence of RAW images

We test the performance of the method with burst sequences and compare it with more adapted techniques for this type of data. We took a burst of a poster pasted on a planar surface. It is well known, that in that case, a parametric tranformation is able to correctly register any two images of the sequence. Such a parametric transformation should include radial distorsion parameters if the two view points are quite different. Most methods estimate a global homography, a tiled translation, or even simpler transformations as a global affinity.

We compare our method with the burst method introduced in [25], which is composed by a global registration with an homography using SIFT [41] characteristic points and a weighted combination of the registered images. This burst method is applied to the images after the imaging chain is applied since we need to compute the characteristic points. We also test the proposed method replacing the optical flow registration in the denoising and demosaicking stages by a global homography registration using SIFT points.

Figure 9 displays the results. The burst algorithm is not able to remove completely noise, since its variance is reduced only as $1/M$ being $M$ the number of frames. However, the denoised result is visually pleasant and has no artifacts. The proposed chain is able to recover all image details and completely removes noise. Our result is not improved by using the global registration, being the two solutions nearly identical. This shows the robustness of our method to the possible inaccuracies of the optical flow registration.
6 Conclusions

We have introduced new denoising and demosaicking methods to deal with a sequence of RAW images. These methods make use of registration by optical flow and robust selection of similar patches by using spatio-temporal distances. The combination of both algorithms yields a sequence free of noise and demosaicking artifacts.

The proposed strategy applies particularly well to image bursts. The comparison with state of the art methods illustrates the performance of the proposed method.

References

[1] D. Alleysson, S. Susstrunk, and J. Herault. Linear demosaicing inspired by the human system. *IEEE Trans. Image Process.*, 14(4):439–449, 2005.

[2] Stefano Andriani, Harald Brendel, Tamara Seybold, and Joseph Goldstone. Beyond the kodak image set: a new reference set of color image sequences. In *Proc. of the IEEE International Conference on Image Processing, Melbourne, Australia*, 2013.

[3] Pablo Arias and Jean-Michel Morel. Towards a bayesian video denoising method. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 107–117. Springer, 2015.

[4] Pablo Arias and Jean-Michel Morel. Video denoising via empirical bayesian estimation of space-time patches. 2017.
Figure 7: Nikon general sequence. From left to right: image formed by directly applying the imaging chain, the video denoising algorithm proposed in [11] and the proposed chain.

[5] B.E. Bayer. Color imaging array, 1976. US Patent 3 971 065.

[6] E.P. Bennett and L. McMillan. Video enhancement using per-pixel virtual exposures. In ACM SIGGRAPH 2005 Papers, page 852. ACM, 2005.

[7] J. Boulanger, C. Kervrann, and P. Bouthemy. Space-time adaptation for patch-based image sequence restoration. IEEE Trans. PAMI, 29(6):1096–1102, 2007.

[8] A. Buades, B. Coll, J.-M. Morel, and C. Sbert. Self-similarity driven color demosaicking. IEEE Trans. Image Process., 18(6):1192–1202, 2009.

[9] A. Buades, B. Coll, and J.M. Morel. A non local algorithm for image denoising. IEEE Computer Vision and Pattern Recognition, 2:60–65, 2005.

[10] A. Buades, J.L. Lisani, and M. Miladinović. Patch Based Video Denoising with Optical Flow Estimation. IEEE Transactions on Image Processing, 25(6):2573–2586, 2016.
Figure 8: Nikon general sequence. From left to right: image formed by directly applying the imaging chain, the video denoising algorithm proposed in [11] and the proposed chain.

[11] Antoni Buades and Jose Luis Lisani. Denoising of noisy and compressed video sequences. In VISIGRAPP (4: VISAPP), pages 150–157, 2017.

[12] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of mathematical imaging and vision, 40(1):120–145, 2011.

[13] Priyam Chatterjee, Neel Joshi, Sing Bing Kang, and Yasuyuki Matsushita. Noise suppression in low-light images through joint denoising and demosaicing. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 321–328. IEEE, 2011.

[14] Miguel Colom, Antoni Buades, and Jean-Michel Morel. Nonparametric noise estimation method for raw images. JOSA A, 31(4):863–871, 2014.

[15] K. Dabov, A. Foi, and K. Egiazarian. Video denoising by sparse 3D transform-domain collaborative filtering. In Proc. 15th European Signal Processing Conference, pages 145–149, Poznan, Poland, 2007.
Figure 9: Image burst sequence. From left to right: image formed by directly applying the imaging chain, the burst fusion [28], the proposed chain and the proposed chain replacing all optical flow registration by a global homography estimation.

[16] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, et al. Bm3d image denoising with shape-adaptive principal component analysis. Proc. of the Workshop on Signal Processing with Adaptive Sparse Structured Representations, Saint-Malo, France, April 2009.

[17] Jingjing Dai, Oscar C Au, Chao Pang, and Feng Zou. Color video denoising based on combined interframe and intercolor prediction. IEEE Transactions on Circuits and Systems for Video Technology, 23(1):128–141, 2013.

[18] Jingjing Dai, Oscar C Au, Feng Zou, and Chao Pang. Generalized multihypothesis motion compensated filter for grayscale and color video denoising. Signal processing, 93(1):70–85, 2013.

[19] Joan Duran and Antoni Buades. Self-similarity and spectral correlation adaptive algorithm for color demosaicking. IEEE transactions on image processing, 23(9):4031–4040, 2014.

[20] Frédo Durand and Julie Dorsey. Fast bilateral filtering for the display of high-dynamic-range images. In ACM transactions on graphics (TOG), volume 21, pages 257–266. ACM, 2002.

[21] M. Ebrahimi and E. R. Vrscay. Multi-frame super-resolution with no explicit motion estimation. In Int. Conf. Image Process. Comput. Vis. Pattern Recogn. (IPCV), pages 455–459, Las Vegas, NV, USA, 2008.

[22] S. Farsiu, M. Elad, and P. Milanfar. Multiframe demosaicing and super-resolution of color images. IEEE Trans. Image Process., 15(1):141–159, 2006.
[23] Yuanyuan Gao, Hai-Miao Hu, and Jiawei Wu. Video denoising algorithm via multi-scale joint luma-chroma bilateral filter. In 2015 Visual Communications and Image Processing (VCIP), pages 1–4. IEEE, 2015.

[24] Murat Gevrekci, Bahadir K Gunturk, and Yucel Altunbasak. Pocs-based restoration of bayer-sampled image sequences. In Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on, volume 1, pages I–753. IEEE, 2007.

[25] Tomomasa Gotoh and Masatoshi Okutomi. Direct super-resolution and registration using raw cfa images. In Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, volume 2, pages II–II. IEEE, 2004.

[26] B.K. Gunturk, Y. Altunbasak, and R.M. Mersereau. Color plane interpolation using alternating projections. IEEE Trans. Image Process., 11(9):997–1013, 2002.

[27] J.F. Hamilton and J.E. Adams. Adaptive color plan interpolation in single sensor color electronic camera, 1997. US Patent 5 629 734.

[28] G. Haro, A. Buades, and J.-M. Morel. Photographing paintings by image fusion. SIAM Journal on Imaging Sciences, 5(3):1055–1087, 2012.

[29] Samuel W Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T Barron, Florian Kainz, Jiawen Chen, and Marc Levo. Burst photography for high dynamic range and low-light imaging on mobile cameras. ACM Transactions on Graphics (TOG), 35(6):192, 2016.

[30] Felix Heide, Markus Steinberger, Yun-Ta Tsai, Mushfiqur Rouf, Dawid Pajak, Dikpal Reddy, Orazio Gallo, Jing Liu, Wolfgang Heidrich, Karen Egiazarian, et al. Flexisp: A flexible camera image processing framework. ACM Transactions on Graphics (TOG), 33(6):231, 2014.

[31] Hui Ji, Sibin Huang, Zuowei Shen, and Yuhong Xu. Robust video restoration by joint sparse and low rank matrix approximation. SIAM Journal on Imaging Sciences, 4(4):1122–1142, 2011.

[32] Hui Ji, Chaqiang Liu, Zuowei Shen, and Yuhong Xu. Robust video denoising using low rank matrix completion. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 1791–1798. IEEE, 2010.

[33] Ljubomir Jovanov, Hiêp Luong, Tijana Ružić, and Wilfried Philips. Multiview image sequence enhancement. In SPIE/IS&T Electronic Imaging, pages 93990K–93990K. International Society for Optics and Photonics, 2015.

[34] Barry K Karch and Russell C Hardie. Robust super-resolution by fusion of interpolated frames for color and grayscale images. Frontiers in Physics, 3:28, 2015.

[35] Minjae Kim, Dubok Park, David K Han, and Hanseok Ko. A novel approach for denoising and enhancement of extremely low-light video. IEEE Transactions on Consumer Electronics, 61(1):72–80, 2015.

[36] M. Lebrun, A. Buades, and J.-M. Morel. A nonlocal bayesian image denoising algorithm. SIAM Journal on Imaging Sciences, 6(3):1665–1688, 2013.

[37] Hwea Yee Lee, Wai Lam Hoo, and Chee Seng Chan. Color video denoising using epitome and sparse coding. Expert Systems with Applications, 42(2):751–759, 2015.
[38] J. Li, C. Bai, Z. Liu, and J. Yu. Optimized color filter arrays for sparse representation-based demosaicking. *IEEE Trans. Image Process.*, 26(5):2381–2393, 2017.

[39] C. Liu and W.T. Freeman. A high-quality video denoising algorithm based on reliable motion estimation. In *European Conference on Computer Vision*, pages 706–719. Springer, 2010.

[40] Ziwei Liu, Lu Yuan, Xiaou Tang, Matt Uyttendaele, and Jian Sun. Fast burst images denoising. *ACM Transactions on Graphics (TOG)*, 33(6):232, 2014.

[41] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[42] Rastislav Lukac and Konstantinos N Plataniotis. Adaptive spatiotemporal video demosaicking using bidirectional multistage spectral filters. *IEEE Transactions on Consumer Electronics*, 52(2):651–654, 2006.

[43] M. Maggioni, G. Boracchi, A. Foi, and K. Egiazarian. Video denoising using separable 4D nonlocal spatiotemporal transforms. In *IS&T/SPIE Electronic Imaging*, pages 787003–787003. International Society for Optics and Photonics, 2011.

[44] Julien Mairal, Guillermo Sapiro, and Michael Elad. Learning multiscale sparse representations for image and video restoration. *Multiscale Modeling & Simulation*, 7(1):214–241, 2008.

[45] D. Menon, S. Andriani, and G. Calvagno. Demosaicing with directional filtering and a posteriori decision. *IEEE Trans. Image Process.*, 16(1):132–141, 2007.

[46] D. Menon and G. Calvagno. Color image demosaicking: An overview. *Signal Process. - Image*, 26(8-9):518–533, 2011.

[47] Ben Mildenhall, Jonathan T Barron, Jiawen Chen, Dillon Sharlet, Ren Ng, and Robert Carroll. Burst denoising with kernel prediction networks. *arXiv preprint arXiv:1712.02327*, 2017.

[48] Mehmet K Ozkan, M Ibrahim Sezan, and A Murat Tekalp. Adaptive motion-compensated filtering of noisy image sequences. *IEEE transactions on circuits and systems for video technology*, 3(4):277–290, 1993.

[49] Dmitriy Paliy, Alessandro Foi, Radu Bilcu, and Vladimir Katkovnik. Denoising and interpolation of noisy bayer data with adaptive cross-color filters. In *Visual Communications and Image Processing 2008*, volume 6822, page 68221K. International Society for Optics and Photonics, 2008.

[50] Sukanya Patil and Ajit Rajwade. Poisson noise removal for image demosaicing. In *BMVC*, 2016.

[51] N. N. Ponomarenko, V. V. Lukin, M. S. Zriakhov, A. Kaarna, and J. T. Astola. An automatic approach to lossy compression of AVIRIS images. *IEEE International Geoscience and Remote Sensing Symposium*, 2007.

[52] M. Protter, M. Elad, H. Takeda, and P. Milanfar. Generalizing the nonlocal-means to super-resolution reconstruction. *IEEE Trans. Image Process.*, 18(1):36–51, 2009.

[53] Matan Protter and Michael Elad. Image sequence denoising via sparse and redundant representations. *IEEE Transactions on Image Processing*, 18(1):27–35, 2009.
[54] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In Computer Vision, 1998. Sixth International Conference on, pages 839–846, 1998.

[55] Patrick Vandewalle, Karim Krichane, David Alleysson, and Sabine Süssstrunk. Joint demosaicing and super-resolution imaging from a set of unregistered aliased images. In Digital Photography III, volume 6502, page 65020A. International Society for Optics and Photonics, 2007.

[56] Bihan Wen, Saiprasad Ravishankar, and Yoram Bresler. Video denoising by online 3d sparsifying transform learning. In Image Processing (ICIP), 2015 IEEE International Conference on, pages 118–122. IEEE, 2015.

[57] Xiaolin Wu and Lei Zhang. Temporal color video demosaicking via motion estimation and data fusion. IEEE Transactions on Circuits and Systems for Video Technology, 16(2):231–240, 2006.

[58] Q. Xu, H. Jiang, R. Scopigno, and M. Sbert. A new approach for very dark video denoising and enhancement. In 17th IEEE International Conference on Image Processing, pages 1185–1188. IEEE, 2010.

[59] Huanjing Yue, Xiaoyan Sun, Jingyu Yang, and Feng Wu. Image denoising by exploring external and internal correlations. IEEE Transactions on Image Processing, 24(6):1967–1982, 2015.

[60] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime TV-L1 optical flow. In Proc. DAGM Symp., volume 4713 of Lecture Notes in Comp. Sci., pages 214–223, Heidelberg, Germany, 2007.

[61] L. Zhang and X. Wu. Color demosaicking via directional linear minimum mean square-error estimation. IEEE Trans. Image Process., 14(12):2167–2178, 2005.

[62] Lei Zhang, Weisheng Dong, Xiaolin Wu, and Guangming Shi. Spatial-temporal color video reconstruction from noisy cfa sequence. IEEE transactions on circuits and systems for video technology, 20(6):838–847, 2010.

[63] Lei Zhang, Weisheng Dong, David Zhang, and Guangming Shi. Two-stage image denoising by principal component analysis with local pixel grouping. Pattern Recognition, 43(4):1531–1549, 2010.