CFA-aware Features for Steganalysis of Color Images

Miroslav Goljan and Jessica Fridrich

Department of ECE, SUNY Binghamton, NY, USA, {mgoljan,fridrich}@binghamton.edu

ABSTRACT

Color interpolation is a form of upsampling, which introduces constraints on the relationship between neighboring pixels in a color image. These constraints can be utilized to substantially boost the accuracy of steganography detectors. In this paper, we introduce a rich model formed by 3D co-occurrences of color noise residuals split according to the structure of the Bayer color filter array to further improve detection. Some color interpolation algorithms, AHD and PPG, impose pixel constraints so tight that extremely accurate detection becomes possible with merely eight features eliminating the need for model richification. We carry out experiments on non-adaptive LSB matching and the content-adaptive algorithm WOW on five different color interpolation algorithms. In contrast to grayscale images, in color images that exhibit traces of color interpolation the security of WOW is significantly lower and, depending on the interpolation algorithm, may even be lower than non-adaptive LSB matching.

1. MOTIVATION

Considering the great advances in steganalysis of grayscale images, it is rather surprising that steganalysis that uses the more complex structure of color images has largely been neglected by the research community. Besides specialized steganalysis techniques that were developed specifically for palette images or images with a low color depth,[4,6,11,19] and techniques focused on a specific embedding paradigm, such as LSB replacement[14] or LSB matching,[9,12] the prior art consists of a handful of techniques whose aim is wide enough to be applicable for detection of modern content-adaptive schemes in color images. The first general-purpose steganalysis feature set for color images that considered dependencies among color channels was proposed by Lyu et al.[18] The authors used higher-order moments of noise residuals obtained using predictors of coefficients in a QMF decomposition of the image from all three color channels.

Recently, the authors of this paper proposed a spatio-color rich model[8] consisting of two parts – the spatial rich model with a single quantization step equal to $q = 1$ (SRMQ1)[7] computed from the union of all three channels and the color rich model (CRMQ1) formed by 3D co-occurrences of residuals taken across the color channels rather than along the spatial directions as in SRMQ1. In images that exhibit traces of color interpolation, the authors discovered that the CRMQ1 is much more effective for detection than the SRMQ1. This was attributed to the fact that the relationship among colors introduced by demosaicking becomes stronger than the dependencies among neighboring pixels that typically occur in natural images.

In this paper, we describe a further extension of the CRM that is aware of the underlying spatial alignment of the Bayer color filter array (CFA). These new CFA-aware features boost the detection in color images that exhibit traces of color interpolation, and this holds true for both non-adaptive and content-adaptive steganography. In order to apply the new feature set, the training as well as the testing needs to be carried out on images that were spatially synchronized w.r.t. the configuration of the Bayer CFA. By this we mean that all images in both the training and the testing set need to be cropped if necessary by at most one pixel in each direction to make sure that, say, the upper left corner corresponds to a non-interpolated blue channel. Since the configuration of the Bayer CFA can be estimated from a single image using existing techniques,[3,13] it is feasible to assume that the steganalyst can spatially synchronize the training database as well as the tested image. The errors associated with this synchronization will of course impact the steganalysis detection errors to a degree that depends on the strength of the color interpolation artifacts as well as the RAW-to-RGB convertor. In general, making the features CFA aware will improve detection only if the Bayer CFA configuration can be estimated reliably.

We describe three versions of the CFA-aware CRM features and test their effectiveness in detecting non-adaptive LSB matching and the content-adaptive algorithm WOW.[10] The versions differ in how the 3D co-occurrences are split based on the type of the pixel w.r.t. Bayer CFA and whether they are directionally symmetrized. This gives the feature sets different dimensionalities and different power to detect steganography. We also study the effect of four different color interpolation algorithms (AHD, PPG, bilinear, and VNG) available in the conversion utility 'ufraw', which incorporates 'dcraw', and the RAW-to-RGB converter in Adobe Lightroom. As one can expect, the interpolation algorithm has a major effect on the detection power. For real life applications, it will be necessary to first inspect a given image and determine at least the "class" of the color interpolation algorithm (and possibly the RAW-to-RGB convertor) and send it to a classifier trained for that particular class. A study like this would require an immense effort involving an exhaustive list of RAW-to-RGB convertors with various color interpolation algorithms, which is well beyond the scope of this paper.

While for images processed using bilinear, VNG, and Lightroom algorithms feature richification is necessary to obtain accurate detection because all features are rather weak, for images outputted by ufraw with the AHD and PPG color interpolation algorithms, quite surprisingly there are about *eight* co-occurrence bins in the 'minmax41c' submodel of the CFA-aware CRM that carry *all the detection power*. We call them the "violator bins" because they are almost empty in cover images and get populated by embedding. This is reminiscent of the JPEG compatibility attack on steganography in decompressed JPEG images.[2,5,15,17] One can also say that images generated by ufraw's AHD or PPG interpolation form a singular source with easily characterizable artifacts (constraints) that can be extremely efficiently used for steganalysis. These two image sources are thus poor choices for a cover source and should be avoided for steganography for the same reason why decompressed JPEG images should not be used for spatial-domain steganography.

In Section 2, we describe the color rich model (CRM) as it forms the basis of its extended version that is aware of the configuration of the Bayer CFA. In the following Section 3, several versions of the CFA-aware CRM are introduced based on how the higher-order statistics in the form of 3D co-occurrences are split and symmetrized. All experimental results with rich models appear in Section 4, where we also evaluate the effectiveness of the proposed rich models and the benefit of making the features aware of the CFA. We also provide the results for all color interpolation algorithms and both steganographic methods. In Section 5, we analyze one specific submodel of the rich model called 'minmax41c'. Using greedy forward feature selection, we identify eight bins that hold all the detection power for images processed using the AHD and PPG algorithms. After explaining why they work so well, we show that, for these two sources, richification is not needed and WOW becomes less secure than non-adaptive LSB matching. The paper is concluded in Section 6, where we also discuss possible future directions.

2. THE COLOR RICH MODEL (CRM)

Here, we describe the color rich model as introduced in Ref. [8]. It is formed by 3D co-occurrences of the noise residuals used in the spatial rich model.[7] We refrain from providing a detailed description of all residuals as this can be found in the above cited reference. Instead, we point out the main differences on specific examples hoping that a reader familiar with the SRM will be able to grasp the CRM as well as its different CFA-aware versions introduced in the next section.

Let us assume that we have a true-color image $\mathbf{I}$ whose color channels are sampled at 8 bits. Each color channel is represented with an $n_1 \times n_2$ matrix of 8-bit non-negative integers: $\mathbf{I} = \{\mathbf{R}, \mathbf{G}, \mathbf{B}\}$, $\mathbf{R} = (r_{ij})$, $\mathbf{G} = (g_{ij})$, $\mathbf{B} = (b_{ij})$, $r_{ij}, g_{ij}, b_{ij} \in \{0, 1, \ldots, 255\}$, $1 \le i \le n_1$, $1 \le j \le n_2$. For simplicity, we will assume that n_1 and n_2 are even.

As an example, we consider a quantized noise residual computed from the red channel using the pixel predictor in the form of the average of its horizontal neighbors (denoted as 'spam12h,v' in Ref. [7]),

$$z_{ij}^{(R,h)} = Q_T\left([r_{ij} - (r_{i,j-1} + r_{i,j+1})/2]/q\right), \tag{1}$$

where $Q_T : \mathbb{R} \to \{-T, -T+1, \ldots, T\}$ is a scalar quantizer with $2T+1$ integer centroids and q is the quantization step. In this paper, we will use a single quantization step $q = 1$ and append 'Q1' to the acronym of the feature name (CRMQ1). Similarly, we also compute the residuals, $z_{ij}^{(G,h)}$, $z_{ij}^{(B,h)}$ from the other two channels. Instead of

forming the co-occurrence from spatially adjacent samples of the residuals as is done in the SRM, in the CRM will form a three-dimensional co-occurrence across the color channels from the triplets $(z_{ij}^{(R,h)}, z_{ij}^{(G,h)}, z_{ij}^{(B,h)})$ for all pixels (i,j):

$$C_{d_1d_2d_3}^{(R,h)} = \sum_{i,j} \left[(z_{ij}^{(R,h)}, z_{ij}^{(G,h)}, z_{ij}^{(B,h)}) = (d_1, d_2, d_3) \right], \quad d_1, d_2, d_3 \in \{-T, \dots, T\}, \tag{2}$$

where $[P]$ denotes the Iverson bracket, which is equal to 1 when the statement P is true and 0 when it is false. Note that since we take the co-occurrences across the color channels, we can add the co-occurrence obtained using the horizontal version of this predictor, $C^{(R,h)}$, and the co-occurrence $C^{(R,v)}$ formed from residuals obtained using the vertical version of the same predictor:

$$z_{ij}^{(R,v)} = Q_T \left([r_{ij} - (r_{i-1,j} + r_{i+1,j})/2]/q \right). \tag{3}$$

We can do this because of symmetries of natural images as the content should not prefer either direction when considered over many natural scenes. Depending on the symmetries of the pixel predictor, we may be adding statistics from four or even eight residual versions after including the mirror versions of the predictor kernels and their versions rotated by 90 degrees (see, e.g., the residuals 'minmax24' and 'minmax48h,v' in Ref. [7]).

The SRM is built from noise residuals of two types – those obtained by linear filtering the image with a fixed kernel (the kernel is $[-1/2 \; 1 \; -1/2]$ in the example above) and those computed as the maximum or minimum of outputs of several linear filters. For example, staying with the red channel, the following is the so-called 'minmax41' residual from the SRM:

$$z_{ij}^{(R,max)} = \max \left\{ z_{i,j+1}^{(R)} - z_{ij}^{(R)}, z_{i,j-1}^{(R)} - z_{ij}^{(R)}, z_{i+1,j}^{(R)} - z_{ij}^{(R)}, z_{i-1,j}^{(R)} - z_{ij}^{(R)} \right\}, \tag{4}$$

$$z_{ij}^{(R,min)} = \min \left\{ z_{i,j+1}^{(R)} - z_{ij}^{(R)}, z_{i,j-1}^{(R)} - z_{ij}^{(R)}, z_{i+1,j}^{(R)} - z_{ij}^{(R)}, z_{i-1,j}^{(R)} - z_{ij}^{(R)} \right\}, \tag{5}$$

which is computed as the minimum (maximum) of differences between the central pixel and its four horizontally and vertically adjacent neighbors.

The co-occurrences can be further compacted using sign and directional symmetries to make them better populated and more robust statistical descriptors. For residuals obtained using linear filters, applying a sign symmetry means that one merges the co-occurrence bins (d_1, d_2, d_3) and $(-d_1, -d_2, -d_3)$, while for the directional symmetry the bin (d_1, d_2, d_3) is merged with (d_3, d_2, d_1). Since the residuals obtained using linear filters are outputs of high-pass filters, their marginal distribution will be symmetrical about zero and the sign symmetrization is justified. However, because the color channels may exhibit different levels of noisiness due to different gains (white balance) applied to them, their marginal distributions may be different. It is thus a question whether the directional symmetry should be applied to our 3D co-occurrences.

As the marginals of min/max residuals are no longer symmetrical, they need to be sign-symmetrized differently. Just as in the SRM, the sign symmetry is applied by merging the bin (d_1, d_2, d_3) from $C^{(R,min)}$ with $(-d_1, -d_2, -d_3)$ from $C^{(R,max)}$. The application of the directional symmetry again needs to be investigated due to the different properties of the color channels.

3. CFA-AWARE CRM

The computation of the CFA-aware CRM starts by computing the residuals from each color channel as explained for the CRM in the previous section. The 3D co-occurrences, however, are formed separately depending on the position of the pixel w.r.t. the Bayer CFA. Recalling the range of pixel indices i and j, $1 \le i \le n_1$, $1 \le j \le n_2$, assuming the element in the upper left corner of the color noise residual $\mathbf{Z} = (\mathbf{Z}^{(R)}, \mathbf{Z}^{(G)}, \mathbf{Z}^{(B)})$ corresponds to a non-interpolated pixel value in the blue channel, we introduce the following four index sets:

$$\mathcal{I}_B = \{(i,j) | i \text{ odd}, j \text{ odd}\}, \quad \mathcal{I}_{G1} = \{(i,j) | i \text{ odd}, j \text{ even}\},$$
$$\mathcal{I}_{G2} = \{(i,j) | i \text{ even}, j \text{ odd}\}, \quad \mathcal{I}_R = \{(i,j) | i \text{ even}, j \text{ even}\},$$

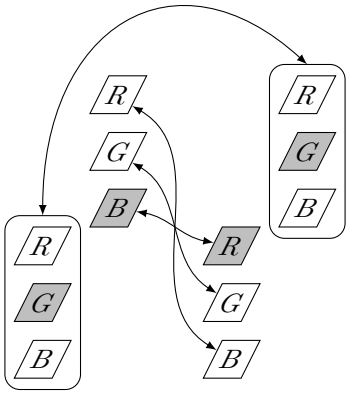

Figure 1. Formation of co-occurrences for the INI/NII split. The diagram shows the three color channels of four adjacent pixels forming a 2×2 square. The shading highlights the measured channels while the unshaded ones are interpolated. Arrows indicate how the residual triples should be merged. The RGB triples at pixels with the green filter are all put into one set from which the 3D co-occurrence $C^{(INI)}$ is formed. The RGB triple at pixels with the red filter are flipped (changed to BGR) before being merged with their corresponding counterparts at pixels with the blue filter to compute $C^{(NII)}$.

and compute four 3D co-occurrences

$$C^{(B)}_{d_1d_2d_3} = \sum_{(i,j)\in\mathcal{I}_B} \big[(z^{(R)}_{ij}, z^{(G)}_{ij}, z^{(B)}_{ij}) = (d_1, d_2, d_3)\big], \quad C^{(G1)}_{d_1d_2d_3} = \sum_{(i,j)\in\mathcal{I}_{G1}} \big[(z^{(R)}_{ij}, z^{(G)}_{ij}, z^{(B)}_{ij}) = (d_1, d_2, d_3)\big], \quad (6)$$

$$C^{(G2)}_{d_1d_2d_3} = \sum_{(i,j)\in\mathcal{I}_{G2}} \big[(z^{(R)}_{ij}, z^{(G)}_{ij}, z^{(B)}_{ij}) = (d_1, d_2, d_3)\big], \quad C^{(R)}_{d_1d_2d_3} = \sum_{(i,j)\in\mathcal{I}_R} \big[(z^{(R)}_{ij}, z^{(G)}_{ij}, z^{(B)}_{ij}) = (d_1, d_2, d_3)\big]. \quad (7)$$

In other words, these four co-occurrences are formed from residual samples corresponding, respectively, to non-interpolated blue, green, green, and red pixels of the Bayer CFA.

We now have several options for symmetrizing and merging these four co-occurrences. We remind the reader that all co-occurrences are always symmetrized by sign. In this work, we investigated the following three options summarized in Table 1.

RB/GG split. Here, we treat $C^{(B)}_{d_1d_2d_3}$ and $C^{(R)}_{d_1d_2d_3}$ as equivalent and apply the directional symmetry to both and then add them. The co-occurrences from green pixels are treated the same way:

$$C^{(RB)}_{d_1d_2d_3} = C^{(B)}_{d_1d_2d_3} + C^{(B)}_{d_3d_2d_1} + C^{(R)}_{d_1d_2d_3} + C^{(R)}_{d_3d_2d_1}, \quad (8)$$

$$C^{(GG)}_{d_1d_2d_3} = C^{(G1)}_{d_1d_2d_3} + C^{(G1)}_{d_3d_2d_1} + C^{(G2)}_{d_1d_2d_3} + C^{(G2)}_{d_3d_2d_1}. \quad (9)$$

R/B/GG split without directional symmetry. In this case, we do not merge the statistics from the red and blue channel but do merge the two green channels. Directional symmetry is not applied to either of the three co-occurrences. Formally, we work with the concatenation of $C^{(R)}_{d_1d_2d_3}$, $C^{(B)}_{d_1d_2d_3}$, and $C^{(G1)}_{d_1d_2d_3} + C^{(G2)}_{d_1d_2d_3}$. This is the most careful split of statistics we consider, and it also has the largest dimensionality.

Feature set	NII/INI	RB/GG	R/B/GG	CRMQ1 ($T = 2$)	CRMQ1 ($T = 3$)
Dir. symmetry	only in $C^{(INI)}$	yes	no	yes	yes
Dimension	5514	4146	10323	2703	5404

Table 1. Three versions of CFA-aware CRM features with their dimensionalities corresponding to $T = 2$. For completeness, we also list the dimensions of the original CRMQ1 set.

NII/INI split. Here, we assume that the triples $(z_{ij}^{(B)}, z_{ij}^{(G)}, z_{ij}^{(R)})$, $(i,j) \in \mathcal{I}_B$ and $(z_{ij}^{(R)}, z_{ij}^{(G)}, z_{ij}^{(B)})$, $(i,j) \in \mathcal{I}_R$ have the same statistical properties and thus can be merged. Additionally, we assume that $C^{(G1)}$ and $C^{(G2)}$ can be symmetrized directionally and added. Formally, for each residual we obtain two co-occurrences

$$C^{(NII)}_{d_1d_2d_3} = C^{(B)}_{d_3d_2d_1} + C^{(R)}_{d_1d_2d_3}, \tag{10}$$

$$C^{(INI)}_{d_1d_2d_3} = C^{(GG)}_{d_1d_2d_3}. \tag{11}$$

The formation of the features is explained graphically in Figure 1. The superscripts relate to the RGB pixel type with 'N' meaning non-interpolated and 'I' meaning interpolated in the RGB triple. The NII pixels correspond to the same set as RB but the two co-occurrences are directionally symmetrized differently.

For all three CFA-aware versions of the CRM, we will use $T = 2$ to keep the overall feature dimensionality comparable to the original CRMQ1. The dimensionalities of all three CFA-aware versions of the CRM are listed in Table 1 together with the dimensionalities of the original CRMQ1 set.

Demosaicking	Determining green filter	Determining red filter
AHD	0.0003	0.0095
VNG	0.0070	0.0070
BIL	0.0000	0.0000
PPG	0.0025	0.0028
LTR	0.0404	0.1468

Table 2. Error rates for the Bayer CFA configuration estimation. Random guessing the configuration would produce values ≈ 0.5 in the first column and ≈ 0.75 in the second column.

4. EXPERIMENTS

In this section, we first describe our experimental setup and then report the results of all experiments with the CFA-aware CRMs. The goal of this study is to evaluate the effectiveness of the three proposed versions of "CFA-awareness" to identify a good compromise between feature dimensionality and the performance. We also evaluate the detection accuracy on five different image sets.

4.1 Image sources and interpolation algorithms

All experiments were executed on a color version of BOSSbase 1.01.[1] The full-resolution raw images were converted using the same script that was used for creating the BOSSbase with the following modifications. The output of ufraw (ver. 0.18 with dcraw ver. 9.06) was changed to the color ppm format instead of the pgm grayscale. Also, we removed the resizing operation and merely cropped the images to their center 514×514 region. We selected the following four demosaicking algorithms in dcraw:

- Patterned Pixel Grouping (PPG),
- Adaptive Homogeneity-Directed interpolation (AHD), which is the default setting,
- Bilinear interpolation (BIL), the basic and the simplest algorithm,
- Threshold-based Variable Number of Gradients interpolation (VNG).

Additionally, we converted the raw images to the true-color TIFF format using Adobe Lightroom 5.6. For the batch export, we selected the default settings, which included the Adobe RGB preset. The database of images converted using Lightroom will be abbreviated LTR.

Since the images in BOSSbase were taken by seven different cameras, we finally synchronized all images in the database by cropping them by one pixel if necessary so that the upper left pixel corresponded to a non-interpolated blue in the Bayer CFA. This way, the final image size was 512×512 pixels.

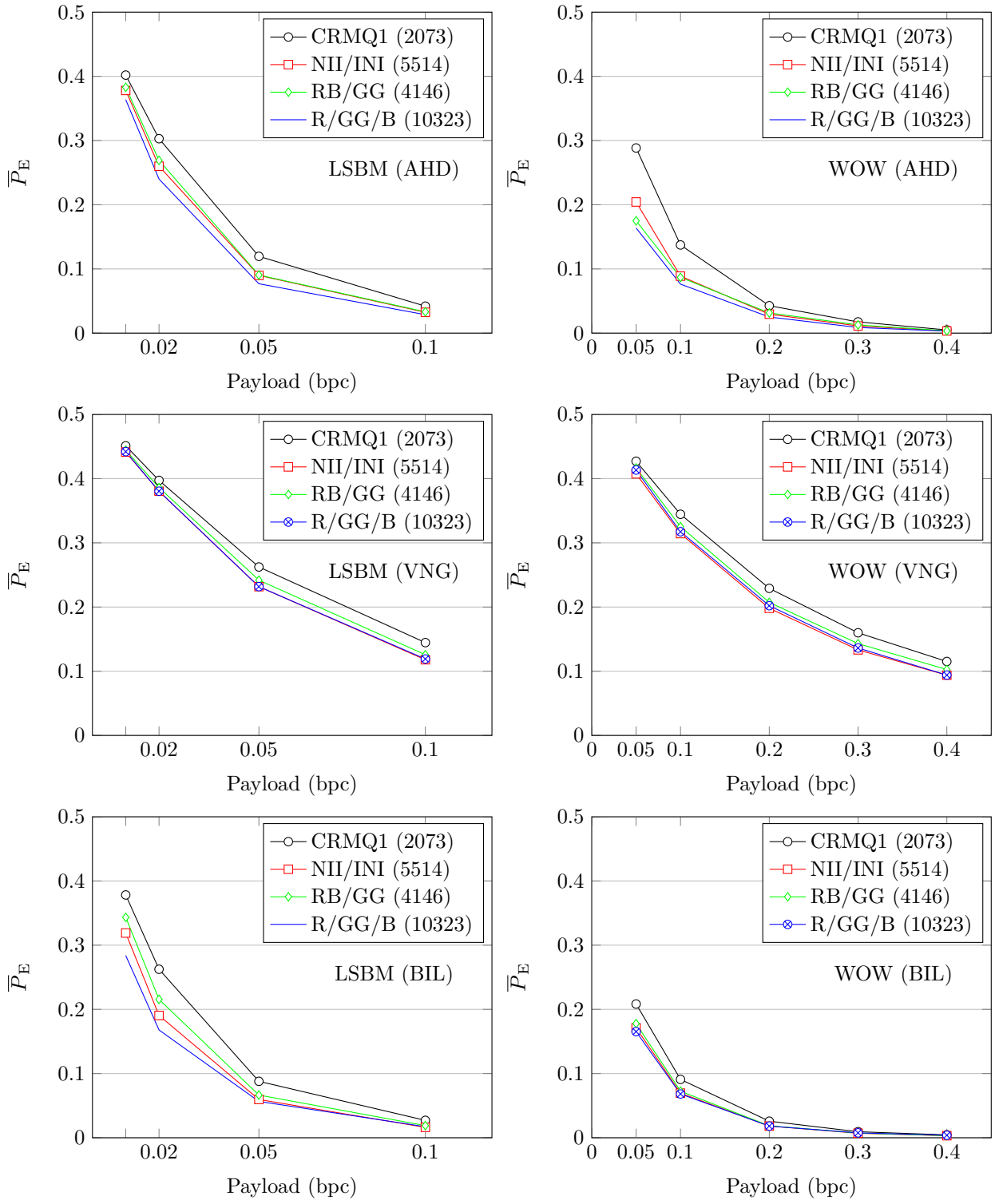

Figure 2. Detection error $\overline{P}_E$ of the four methods as a function of payload for LSBM (left) and WOW (right) on AHD (top), VNG (middle), and BIL (bottom).

Notice that the CFA-aware features with the R/B/GG as well as the NII/INI split require the knowledge of the exact CFA configuration, including the knowledge of which pixels correspond to non-interpolated red and blue. The RB/GG feature set only requires the knowledge of the green pixels. Lee[3] introduced a simple and efficient method for determining the CFA configuration. We tested this method on all five versions of BOSSbase. Table 2 reports the error rate of incorrectly determining the CFA configuration across all five sources. As the ground truth, we took the majority estimated configuration for a given camera and image orientation (landscape vs. portrait). All experiments in this paper were thus executed with both the training and testing images perfectly aligned. In practice, our failure to determine the correct configuration of the CFA will decrease the steganalysis detection accuracy. Considering the low estimation errors in Table 2, this decrease will be negligible at least for the first four demosaicking options. A limited experiment investigating the effect of incorrect estimation of the CFA configuration in test images only is included in Section 5.

4.2 Detectors and tested steganographic schemes

All detectors were trained as binary classifiers implemented using the FLD ensemble[16] with default settings. A separate classifier was trained for each image source, embedding algorithm, and payload to show how the detection performance depends on the payload size. The ensemble by default minimizes the total classification error probability under equal priors $P_{\text{E}} = \min_{P_{\text{FA}}} \frac{1}{2}(P_{\text{FA}} + P_{\text{MD}})$, where P_{FA} and P_{MD} are the false-alarm and missed-detection probabilities. We evaluate the security by averaging P_{E} measured on the testing set over ten 5000/5000 database splits, and denote it as $\overline{P}_{\text{E}}$.

Two embedding algorithms were tested: the non-adaptive ternary LSB matching (LSBM) with the change rate as the distortion measure and the content-adaptive WOW[10] both simulated at their corresponding rate–distortion bounds. Both algorithms were applied to color images by treating their color channels as three grayscale images and embedding the same relative payload in each channel.

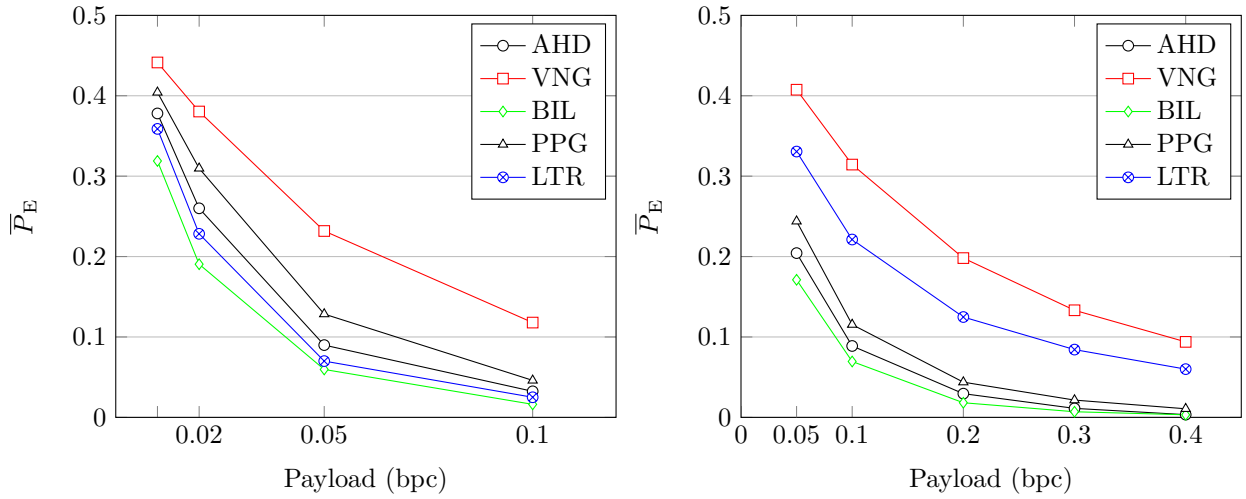

Figure 3. Detection error $\overline{P}_{\text{E}}$ as a function of payload when steganalyzing five differently interpolated versions of color BOSSbase. LSBM (left) and WOW (right).

4.3 Effect of making the CRM aware of the CFA configuration

In this section, we assess the effectiveness of the three CFA-aware versions of the CRM proposed in Section 3. We remind that the dimensionality of the original CRM with the same value of $T = 2$ is $2,073$, which is what we used in this paper as well. Our goal is to select a good compromise between complexity (feature dimensionality) and detection accuracy.

Figure 2 shows the detection error for LSBM and WOW as a function of embedded payload for the original CRMQ1[8] and its three CFA-aware versions for images processed using ufraw with AHD, VNG, and BIL color interpolation methods. We are not showing the results for PPG as they are qualitatively and quantitatively

	Payload (bpp)			
LSBM	0.01	0.02	0.05	0.1
'AHD'	0.3780	0.2600	0.0899	0.0324
'VNG'	0.4414	0.3804	0.2318	0.1179
'BIL'	0.3189	0.1905	0.0596	0.0162
'PPG'	0.4042	0.3097	0.1285	0.0460
'LTR'	0.3588	0.2283	0.0700	0.0250

	Payload (bpp)				
WOW	0.05	0.1	0.2	0.3	0.4
'AHD'	0.2043	0.0888	0.0295	0.0112	0.0034
'VNG'	0.4075	0.3144	0.1981	0.1332	0.0939
'BIL'	0.1711	0.0695	0.0182	0.007	0.0032
'PPG'	0.2439	0.1154	0.0438	0.0215	0.0107
'LTR'	0.3306	0.2212	0.1248	0.0843	0.0600

Table 3. Numerical values of all detection errors from Figure 3.

similar to those of AHD. We are not showing the results for LTR images either because for this source all four feature sets (three CFA aware and CRM) have essentially the same performance. In other words, splitting the co-occurrences does not bring any detection improvement. This is likely due to the fact that the color interpolation artifacts in LTR images are too weak for the split to aid steganalysis (c.f. the accuracy of estimating the CFA configuration in Table 2).

The biggest improvement of CFA-aware versions of the CRMQ1 w.r.t. the original CRMQ1 is for images processed using AHD (and PPG) and BIL. The improvement is especially significant (up to 10% gain in detection error) for small payloads for both LSBM and WOW. For VNG images, the overall detection is lower and the benefit of splitting the statistics is rather small.

When comparing the three versions of CFA-aware features, as expected the largest R/B/GG split provides the best performance in all cases. The RB/GG and NII/INI splits are comparable and also have similar dimensionalities. We selected the NII/INI split for our further experiments.

4.4 Performance across interpolation algorithms

Figure 3 shows the average testing error $\overline{P}_{\mathrm{E}}$ as a function of the relative payload for LSBM (left) and WOW (right) for the CFA-aware CRMQ1 based on the NII/INI split (see Table 1) for five color interpolation methods. The detection error depends strongly on the type of the color interpolation algorithm and the RAW-to-RGB convertor. Images processed using the bilinear interpolation algorithm are the most detectable for both LSBM and WOW. The most difficult images for detection are those processed using the VNG algorithm. Also note the extremely high detectability of steganography in these color images compared to their grayscale counterparts of the same size (BOSSbase 1.01), see, e.g., Ref. [10]. This high detectability is due to the fact that the images are cropped from their full resolution rather than resized and due to the presence of strong dependencies among color channels. For completeness, in Table 3 we provide the numerical values of all detection errors from Figure 3.

5. THE MAGNIFICENT EIGHT

In order to gain more insight into the rich feature sets, we steganalyzed both LSBM and WOW with the individual submodels of the CRMQ1 feature set with the NII/INI split. While for images processed using VNG and LTR, all submodels had a similar performance, for AHD and PPG, there was one extremely strong submodel – the one computed from the 'minmax41' residual (4)–(5). This submodel in BIL images was still the strongest but comparably less powerful than in AHD and PPG images. In the NII/INI split, the 'minmax41' submodel has the dimensionality of 125 in $C^{(NII)}$ and 75 in the $C^{(INI)}$ co-occurrence. We will call this color version of the submodel 'minmax41c' to distinguish it from its grayscale counterpart as introduced in Ref. [7]. To find out if there is a small feature set responsible for the detection power, we applied a greedy forward feature selection to the $125 + 75 = 200$-dimensional feature vector consisting of the concatenation of $C^{(NII)}$ and $C^{(INI)}$. The feature selection first identified the best individual co-occurrence bin, then searched for the next bin that best supplemented the first, etc. The result of this analysis revealed that only about eight bins are responsible for the detection, namely all 2^3 bins (d_1, d_2, d_3) with $d_i \in \{1, 2\}$ corresponding to the NII pixels (pixels with an interpolated green channel).

The reason for their excellent performance can be explained by interpreting the 'minmax41c' residual within the context of the color interpolation. The reader is instructed to follow Figure 4. Assuming the simplest bilinear

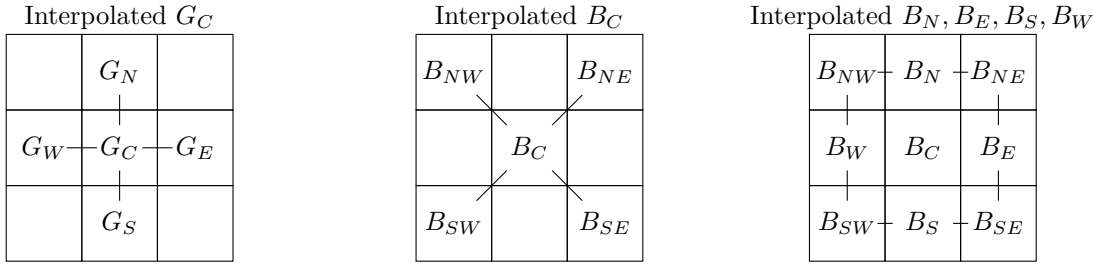

Figure 4. The structure of bilinear interpolation at an NII pixel. The short lines indicate the averaging in bilinear interpolation.

interpolation, the green channel at the central NII pixel is obtained from its four neighbors as their arithmetic average:

$$G_C = \frac{1}{4}(G_S + G_N + G_E + G_W). \tag{12}$$

Thus, it is immediate that the 'minmax41c' residual value at the central green pixel, $\min\{G_S - G, G_N - G, G_E - G, G_W - G\}$, cannot be positive. Moreover, at an NII pixel, the blue channel is interpolated from its four neighbors as

$$B_C = \frac{1}{4}(B_{NE} + B_{SE} + B_{SW} + B_{NW}). \tag{13}$$

The 'minmax41c' residual in the blue channel is $z_C^{(B,min)} = \min\{B_S - B_C, B_N - B_C, B_E - B_C, B_W - B_C\}$. Because in the bilinear interpolation

$$B_S = \frac{1}{2}(B_{SE} + B_{SW}), \quad B_N = \frac{1}{2}(B_{NE} + B_{NW}), \tag{14}$$

$$B_E = \frac{1}{2}(B_{NE} + B_{SE}), \quad B_W = \frac{1}{2}(B_{NW} + B_{SW}), \tag{15}$$

we have

$$B_S - B_C = \frac{1}{2}(B_{SE} + B_{SW}) - \frac{1}{4}(B_{NE} + B_{SE} + B_{SW} + B_{NW}) = \frac{1}{4}(-B_{NE} + B_{SE} + B_{SW} - B_{NW}),$$
$$B_N - B_C = \frac{1}{2}(B_{NE} + B_{NW}) - \frac{1}{4}(B_{NE} + B_{SE} + B_{SW} + B_{NW}) = \frac{1}{4}(B_{NE} - B_{SE} - B_{SW} + B_{NW}),$$
$$B_E - B_C = \frac{1}{2}(B_{NE} + B_{SE}) - \frac{1}{4}(B_{NE} + B_{SE} + B_{SW} + B_{NW}) = \frac{1}{4}(B_{NE} + B_{SE} - B_{SW} - B_{NW}),$$
$$B_W - B_C = \frac{1}{2}(B_{NW} + B_{SW}) - \frac{1}{4}(B_{NE} + B_{SE} + B_{SW} + B_{NW}) = \frac{1}{4}(-B_{NE} - B_{SE} + B_{SW} + B_{NW}).$$

In particular,

$$B_S - B_C = -(B_N - B_C), \tag{16}$$
$$B_E - B_C = -(B_W - B_C), \tag{17}$$

which again prohibits positive values in $z_C^{(B,min)}$.

Therefore, the bins with indices $d_i \in \{1, 2\}$ ($[1,1,1], [2,1,1], [1,2,1], ...$) are essentially "violator bins" that should be (almost) empty in cover images and that can be populated by embedding. In three of our databases, AHD, PPG, and BIL, these bins are completely empty in all covers, and in the other two, VNG and LTR, they are non-empty. The violations in covers can be naturally introduced by the demosaicking algorithm itself as well as by spatial filtering and antialiasing in the particular RAW-to-RGB convertor, and by color correction. The violator bins stand a chance of detecting modifications of pixels by ± 1 only when their neighboring bins, such as $[1,0,1], [2,0,1], [2,0,2], [1,0,2]$, ... , are well populated. While this is true for AHD and PPG databases,

bilinear interpolation, with its simple averaging places the interpolated value further away from the minimum and maximum of the four adjacent pixel values and thus populates the neighboring bins in 'minmax41c' much less. This is why, we believe, the violator bins are less efficient for steganalysis in the BIL database.

The third element in an NII bin comes from a channel that was not interpolated (the red channel). Positive values of the 'minmax41c' residual are, however, still infrequent as they correspond to a local minimum in the red channel. Thus, this bin will also be sensitive to embedding. In fact, we tested the marginals of the above eight bins for detection of steganography (e.g., counting only the violations in the green channel) but their detection was not nearly as accurate as that of the full 3D co-occurrence bins. Apparently, the mutual relationship among the color channels plays an important role for steganalysis.

Figure 5 contrasts the detection error of the NII/INI CFA-aware CRMQ1 with that of the eight violator bins for LSBM and WOW in images processed by AHD and PPG in ufraw. Notice that the eight violator bins outperform the entire rich model. The improvement increases with decreasing payload and is very significant for WOW. In fact, WOW appears more detectable than the non-adaptive LSBM! (Also see the numerical values of detection errors in Table 4.) This is because the content-adaptive WOW clusters the embedding changes in textured regions, which 1) increases the probability of modifying more than one channel at such pixels and 2) increases the probability of modifying the 'minmax41c' noise residuals (differences between neighboring pixels) by more than 1.

We did attempt to force the 8 violator bins into every subspace of the FLD ensemble to see if the additional features in the rich model further improve detection. This step indeed brought a small improvement (about 1%) in LTR images but no improvement was observed for PPG and AHD images.

With steganalysis detection errors of the order of 0.002 (see the errors for large payloads in Table 4), it is natural to ask whether and how much the incorrect estimation of the CFA configuration affects the detection. To obtain some insight, we carried out an experiment with the eight violator bins by training on images whose CFA configuration was perfectly aligned and testing on the remaining images aligned by estimating the CFA configuration for each image individually. In contrast to all our previous experiments, where the alignment was always perfect for both training and testing images, we now incorrectly estimate the configuration in testing images with probabilities listed in Table 4. (We confirm that the stego changes have virtually no effect on the accuracy of estimating the CFA configuration.) We executed this for both the AHD and PPG databases for both steganographic algorithms at 0.4 bpp. This payload was chosen intentionally because the steganalysis detection errors are the lowet for large payloads. For PPG, the detection errors indeed slightly increased from 0.0052 to 0.0060 for LSBM and form 0.0036 to 0.0039 for WOW. For AHD images, the observed increase was no more than 0.0001. In summary, the misestimated CFA configuration affected the detection errors only negligibly as the reported increase is within the statistical spread of the detection errors.

6. CONCLUSIONS

Color interpolation, also called demosaicking, is a form of upsampling. As such, it introduces dependencies among colors and constraints that can be used for detection of steganography. Our current analysis and previous work show that in images that exhibit traces of demosaicking the dependencies between color channels are stronger than the dependencies among adjacent pixels. Thus, it makes sense to build steganalysis features as joint statistics of noise residuals across the color channels rather than spatially-adjacent pixels. When richified, such feature sets are called color rich models, and they are very effective for steganalysis of color images.

In this paper, we investigated whether detection can be further boosted by synchronizing the training and test images by the configuration of the Bayer color filter array and splitting the higher-order statistics (co-occurrences) accordingly. We introduced three different versions of such CFA-aware features and studied their detection performance w.r.t. five different demosaicking algorithms for two steganographic methods LSB matching and WOW. We observed that the detection accuracy as well as the boost from CFA-awareness heavily depends on the demosaicking algorithm and, in general, on the RAW-to-RGB converter. The richification helps with detection of images processed using bilinear and VNG demosaicking in ufraw and using Adobe Lightroom. Except for Lightroom images, the CFA awareness improves detection and this gain is especially significant for small payloads for both LSB matching and WOW.

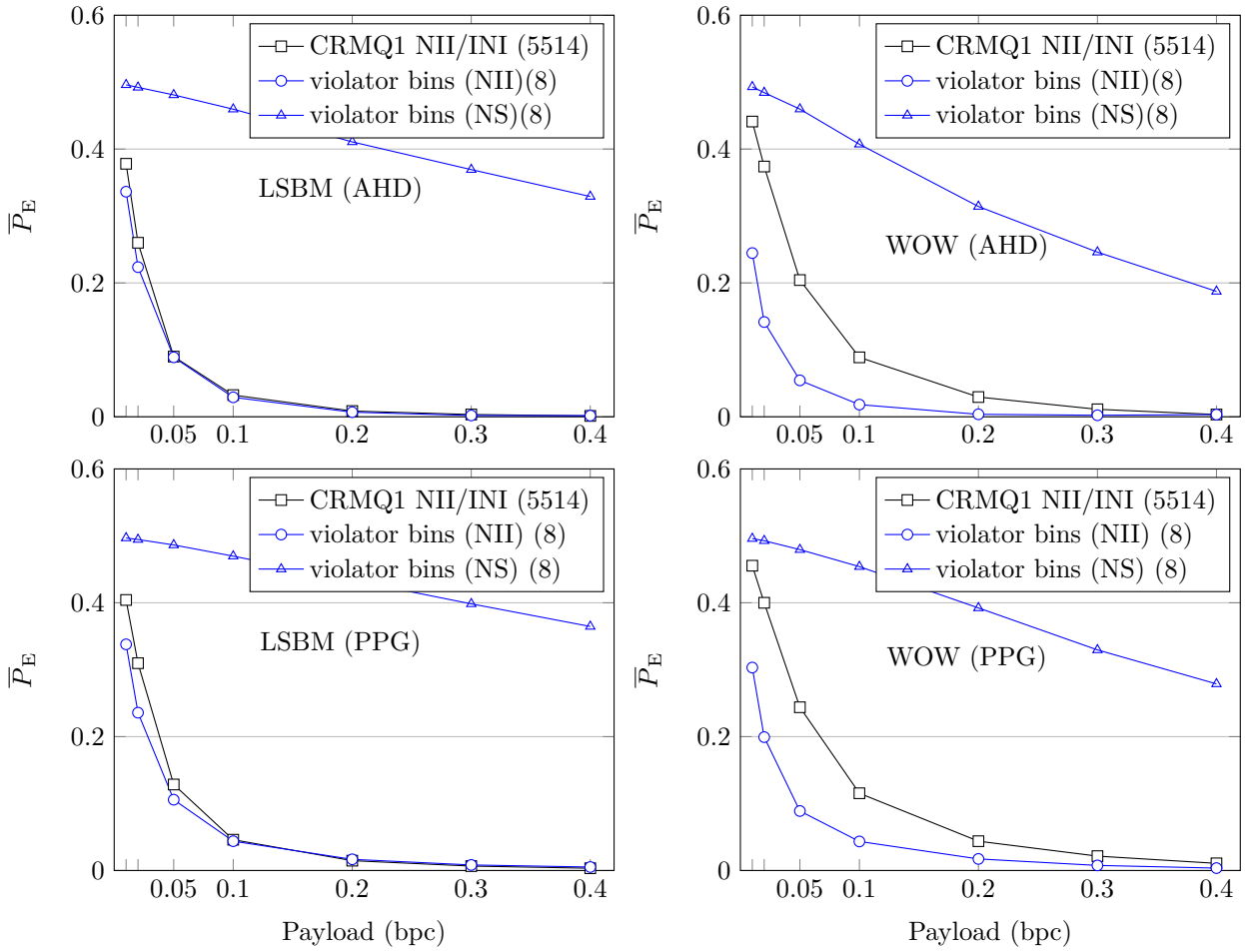

Figure 5. Detection error $\overline{P}_E$ of eight violator co-occurrence bins from interpolated green pixels (NII) taken from the NII/III split of the 'minmax41c' submodel compared to the NII/NII split of the entire CRMQ1. LSBM (left) and WOW (right) on AHD (above) and PPG (below). Also shown is the poor detection ability of the same 8 co-occurrence bins from CRMQ1 when their statistic is not split (NS) according to the CFA.

In images processed using AHD and PPG demosaicking in ufraw, we discovered that the richification is not desirable as there exist eight bins in the 'minmax41c' submodel that hold all the detection power. These bins are violator bins that are nearly empty in cover images but get populated by steganography. This indicates that the two sources are singular (and should be avoided as covers for steganography) in the sense that one can identify a rather simple deterministic "compatibility constraint" and use it to build an extremely accurate detector. The content-adaptive algorithm WOW is especially vulnerable because it concentrates the embedding changes into textured regions, making it more likely to disturb more than one color channel in a given pixel. The steganalysis error of WOW becomes smaller than 1% for payloads as small as 0.2 bits per pixel!

We would like to point out that what affects the detectability of steganography is not only the demosaicking algorithm but the entire processing pipeline executed when converting a RAW image format to a true-color image. Because the convertor in Lightroom is not publicly available, we were not able to determine the reason why CFA awareness did not improve detection for images processed by Lightroom. Interpreting the results is not easy even when using the open-source ufraw (dcraw) because some parts of the conversion are driven by a camera profile and other parts are hidden in a not-so-transparent code.

As a future direction, we list the possibility to build a forensic-aided steganalyzer that would first identify the type of the RAW-to-RGB convertor that was likely used to produce a given test image and then apply a

	Payload (bpp)		0.01	0.02	0.05	0.1	0.2	0.3	0.4
AHD	NII/INI split of CRMQ1	LSBM	0.3780	0.2600	0.0899	0.0324	0.0086	0.0034	0.0013
		WOW	0.4411	0.3739	0.2043	0.0888	0.0295	0.0112	0.0034
	8 violator bins	LSBM	0.3363	0.2236	0.0889	0.0291	0.0069	0.0020	0.0017
		WOW	0.2446	0.1416	0.0544	0.0182	0.0037	0.0023	0.0033
PPG	NII/INI split of CRMQ1	LSBM	0.4042	0.3097	0.1285	0.0460	0.0147	0.0066	0.0034
		WOW	0.4554	0.4000	0.2439	0.1154	0.0438	0.0215	0.0107
	8 violator bins	LSBM	0.3379	0.2358	0.1058	0.0438	0.0167	0.0083	0.0052
		WOW	0.3032	0.1993	0.0889	0.0434	0.0173	0.0076	0.0036

Table 4. Numerical values of all detection errors shown in Figure 5. The statistical spread estimated as a sample standard deviation from ten P_{E} measurements varied form 0.0004 at payload 0.4 bpp to 0.0051 at payload 0.02 bpp.

properly trained classifier. This would not be a small undertaking due to the need to analyze a potentially large number of convertors and their settings as well as color interpolation algorithms.

7. ACKNOWLEDGMENTS

The work on this paper was supported by Air Force Office of Scientific Research under the research grant number FA9950-12-1-0124. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation there on. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of AFOSR or the U.S. Government. The authors would like to thank Rémi Cogranne for help with preparing the images processed with ufraw and useful discussions. Special thanks go to Matthias Kirchner for useful feedback and help with creating pgf graphics in this paper.

REFERENCES

1. P. Bas, T. Filler, and T. Pevný. Break our steganographic system – the ins and outs of organizing BOSS. In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Conference*, volume 6958 of Lecture Notes in Computer Science, pages 59–70, Prague, Czech Republic, May 18–20, 2011.
2. R. Böhme. *Advanced Statistical Steganalysis*. Springer-Verlag, Berlin Heidelberg, 2010.
3. Chang-Hee Choi, Jung-Ho Choi, and Heung-Kyu Lee. CFA pattern identification of digital cameras using intermediate value counting. In J. Dittmann, S. Craver, and C. Heitzenrater, editors, *Proceedings of the 13th ACM Multimedia & Security Workshop*, pages 21–26, Niagara Falls, NY, September 29–30, 2011.
4. J. Fridrich, R. Du, and M. Long. Steganalysis of LSB encoding in color images. *IEEE International Conference on Multimedia and Expo*, 3:1279–1282, 2000.
5. J. Fridrich, M. Goljan, and R. Du. Steganalysis based on JPEG compatibility. In A. G. Tescher, editor, *Special Session on Theoretical and Practical Issues in Digital Watermarking and Data Hiding, SPIE Multimedia Systems and Applications IV*, volume 4518, pages 275–280, Denver, CO, August 20–24, 2001.
6. J. Fridrich, M. Goljan, and D. Soukal. Higher-order statistical steganalysis of palette images. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents V*, pages 178–190, Santa Clara, CA, January 21–24, 2003.
7. J. Fridrich and J. Kodovský. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3):868–882, June 2011.
8. M. Goljan, R. Cogranne, and J. Fridrich. Rich model for steganalysis of color images. In *Sixth IEEE International Workshop on Information Forensics and Security*, Atlanta, GA, December 3–5, 2014.
9. J. J. Harmsen and W. A. Pearlman. Steganalysis of additive noise modelable information hiding. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents V*, volume 5020, pages 131–142, Santa Clara, CA, January 21–24, 2003.
10. V. Holub and J. Fridrich. Designing steganographic distortion using directional filters. In *Fourth IEEE International Workshop on Information Forensics and Security*, Tenerife, Spain, December 2–5, 2012.

11. N. F. Johnson and S. Jajodia. Steganalysis of images created using current steganography software. In D. Aucsmith, editor, *Information Hiding, 2nd International Workshop*, volume 1525 of Lecture Notes in Computer Science, pages 273–289, Portland, OR, April 14–17, 1998. Springer-Verlag, New York.
12. A. D. Ker. Resampling and the detection of LSB matching in color bitmaps. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VII*, volume 5681, pages 1–15, San Jose, CA, January 16–20, 2005.
13. M. Kirchner. Efficient estimation of CFA pattern configuration in digital camera images. In N. D. Memon, A. Alattar, E. J. Delp, and J. Dittmann, editors, *Proceedings SPIE, Electronic Imaging, Media Forensics and Security II*, volume 7541, pages 11 1–12, San Jose, CA, January 17–21, 2010.
14. M. Kirchner and R. Böhme. Steganalysis in technicolor: Boosting WS detection of stego images from CFA-interpolated covers. In *Proc. IEEE ICASSP*, Florence, Italy, May 4–9, 2014.
15. J. Kodovský and J. Fridrich. JPEG-compatibility steganalysis using block-histogram of recompression artifacts. In M. Kirchner and D. Ghosal, editors, *Information Hiding, 14th International Conference*, volume 7692 of Lecture Notes in Computer Science, pages 78–93, Berkeley, California, May 15–18, 2012.
16. J. Kodovský, J. Fridrich, and V. Holub. Ensemble classifiers for steganalysis of digital media. *IEEE Transactions on Information Forensics and Security*, 7(2):432–444, 2012.
17. W. Luo, Y. Wang, and J. Huang. Security analysis on spatial ±1 steganography for JPEG decompressed images. *IEEE Signal Processing Letters*, 18(1):39–42, 2011.
18. S. Lyu and H. Farid. Steganalysis using color wavelet statistics and one-class support vector machines. In E. J. Delp and P. W. Wong, editors, *Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VI*, volume 5306, pages 35–45, San Jose, CA, January 19–22, 2004.
19. A. Westfeld. Detecting low embedding rates. In F. A. P. Petitcolas, editor, *Information Hiding, 5th International Workshop*, volume 2578 of Lecture Notes in Computer Science, pages 324–339, Noordwijkerhout, The Netherlands, October 7–9, 2002. Springer-Verlag, Berlin.