ROBUST CORRECTION STEP FOR CFA INTERPOLATION SCHEMES

Rastislav Lukac[1] *Karl Martin*[1] *Konstantinos N. Plataniotis*[1] *Bogdan Smolka*[2]

[1] Bell Canada Multimedia Laboratory, The Edward S. Rogers Sr. Department of ECE, University of Toronto, 10 King's College Road, Toronto, M5S 3G4 Ontario, Canada
lukacr@ieee.org, kmartin@dsp.utoronto.ca, kostas@dsp.utoronto.ca
[2] Department of Automatic Control, Silesian University of Technology, Akademicka 16 Str., 44-101 Gliwice, Poland, *bsmolka@ia.polsl.gliwice.pl*

ABSTRACT

This paper presents an efficient postprocessing/correction method capable of reducing visual artifacts introduced during the color filter array interpolation process. Edge-sensing weights and the original color filter array data are used to detect structural elements in the original image and correct previously interpolated color components accordingly. The method produces excellent results in terms of both objective and subjective image quality measures.

1. INTRODUCTION

Color filter array (CFA) interpolation or demosaicking is a required step in single-sensor imaging solutions. The Bayer pattern [2], the most popular CFA solution, is commonly used in image-enabled wireless phones, pocket devices and visual sensors for surveillance and automotive applications. Since only a single color component is available at each spatial position of the CFA output, the restored color image is obtained by interpolating the missing color components from the spatially adjacent CFA data.

Most of the available demosaicking designs introduce visual artifacts in the form of blurred edges and false colors [7],[11],[15]. Therefore, demosaicked color image postprocessing techniques, implemented either directly at the hardware level or as an additive software module, should be used to reduce visual impairments introduced during CFA interpolation [12]. The proposed method accepts a full color restored image as input and corrects the previously interpolated components by way of the original CFA data (not affected by interpolation and thus present in the interpolated image) and edge-sensing weights. This can be treated as either a correction procedure incorporated as the last step in a demosaicking scheme or as an independent postprocessing operation.

2. PROPOSED POSTPROCESSING/CORRECTION STEP

Since there is no method to objectively determine whether or not a color component is inaccurate, the proposed postprocessing framework utilizes the differences between the interpolated color components and the original Bayer CFA data included in the restored color output to complete the task [12].

Let us consider a $K_1 \times K_2$ color image $\mathbf{x}(p,q) : Z^2 \rightarrow Z^3$ representing a two-dimensional matrix of three-component color samples $\mathbf{x}_{(p,q)} = (x_{(p,q)1}, x_{(p,q)2}, x_{(p,q)3}) \in Z^3$. Assuming that $p = 1, 2, ..., K_1$ and $q = 1, 2, ..., K_2$ denote the spatial position of a pixel in the vertical (image rows) and horizontal (image columns) directions respectively, the restored color image $\mathbf{x}(p,q)$ is obtained by a conventional Bayer CFA pattern (*Fig.1*) based demosaicking scheme using the original R and B CFA components located at $(\text{odd } p, \text{ even } q)$ and $(\text{even } p, \text{ odd } q)$ respectively, and original G components in the rest of the locations.

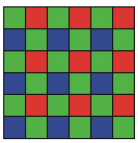

Fig. 1. Bayer CFA pattern [2].

Employing the spectral correlation between the G and R (or B) components of the restored, full color image $\mathbf{x}(p,q)$ into the proposed correction process further improves the color appearance in the rest of $\mathbf{x}(p,q)$. Based on the color-difference model of [1], which is widely utilized in demosaicking [5],[6],[9],[13],[14], the proposed postprocessing scheme re-evaluates the interpolated G components $x_{(p,q)2}$ as follows:

$$x_{(p,q)2} = x_{(p,q)k} + \frac{\sum_{(i,j)\in\zeta} w_{(i,j)}(x_{(i,j)2} - x_{(i,j)k})}{\sum_{(i,j)\in\zeta} w_{(i,j)}} \qquad (1)$$

where $\zeta = \{(p-1,q),(p,q-1),(p,q+1),(p+1,q)\}$ denotes the spatial locations of the original G CFA components surrounding the interpolated location (p,q). As can be seen in *Fig.2a*, the original G CFA components form a diamond-shape mask on the image lattice. The values $w_{(i,j)}$ are edge-sensing weights, $x_{(p,q)k}$ denotes the original R (or B) component at the position being considered, and $(x_{(i,j)2} - x_{(i,j)k})$ denotes the differences between the surrounding original G components and the previously interpolated R (or B) outputs. If (p,q) is a position which corresponds to a Bayer pattern R component, then $k = 1$ is used. Otherwise the position corresponds to a Bayer pattern B component and $k = 3$.

The color difference $(x_{(i,j)2} - x_{(i,j)k})$ used in (1) exhibits a reduction in high-frequency components compared to the individual color planes. Thus, the averaging operation in (1) results in a smaller estimation error compared to the error produced by an interpolator operating on the original R or B components. The utilization of $x_{(p,q)k}$ in (1) preserves the high frequency components of the original R and B color channels since the addition of $x_{(p,q)k}$ to the normalized weighted sum scales the interpolator's output to the desired intensity range.

0-7803-8554-3/04/$20.00 ©2004 IEEE.

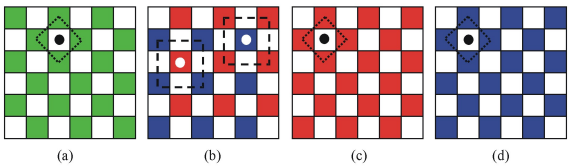

Fig. 2. Spatial arrangements of the pixels and shape-masks obtained during the proposed postprocessing.

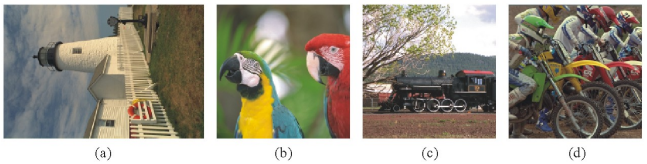

Fig. 3. Test color images: (a) Lighthouse, (b) Parrots, (c) Train, (d) Bikes.

Each $x_{(i,j)2}$ is associated with a positive, real-valued, edge-sensing weight $w_{(i,j)}$ defined as $w_{(i,j)} = [1 + f(d_{(i,j)})]^{-1}$, where $f(\cdot)$ is a function of the aggregated absolute difference $d_{(i,j)} = \sum_{(g,h)\in\zeta} |x_{(i,j)k} - x_{(g,h)k}|$ between the CFA inputs $x_{(i,j)k}$ and $x_{(g,h)k}$. Since the k in $x_{(i,j)k}$ should correspond to the interpolated output, G inputs $x_{(i,j)2}$ are used to determine $w_{(i,j)}$ in (1).

The actual shape of the function $f(d_{(i,j)})$ determines the properties of the weights. In this work, the weights in (1) are implemented using the following form:

$$w_{(i,j)} = [1 + d_{(i,j)}]^{-1} \tag{2}$$

The weighting coefficients $w_{(i,j)}$, for $(i,j) \in \zeta$, reflect the accuracy of the input components $x_{(i,j)k}$ with respect to the Bayer data structural content. When no edge is positioned across the considered directions, the corresponding aggregated absolute difference $d_{(i,j)}$ is small and the CFA component $x_{(i,j)k}$ is not penalized. If the $x_{(i,j)k}$ and $x_{(g,h)k}$ used in calculating $d_{(i,j)}$ are located across an edge, the corresponding absolute difference $d_{(i,j)}$ increases and the corresponding $x_{(i,j)k}$ is appropriately penalized via $w_{(i,j)}$ of (2).

Through the normalization procedure of (1), two constraints necessary to ensure that the output $x_{(p,q)2}$ is an unbiased solution are satisfied. Namely, i) each weight is a positive number, $w_{(i,j)} \geq 0$, and ii) the summation of all the weights is equal to unity $\sum_{(i,j)\in\zeta} \left(w_{(i,j)} / \sum_{(g,h)\in\zeta} w_{(g,h)} \right)$.

The utilization of the corrected G components obtained via (1) for the correction of the interpolated R and B components improves contrast in both the R and B channels of the restored color image $\mathbf{x}(p,q)$. This correction step completes the proposed postprocessing scheme. First, the R (or B) components on positions with original B (or R) CFA data are corrected as follows:

$$x_{(p,q)k} = x_{(p,q)2} + \frac{\sum_{(i,j)\in\zeta} w_{(i,j)}(x_{(i,j)k} - x_{(i,j)2})}{\sum_{(i,j)\in\zeta} w_{(i,j)}} \tag{3}$$

where $\zeta = \{(p-1, q-1), (p-1, q+1), (p+1, q-1), (p+1, q+1)\}$ and k denotes either the R ($k = 1$) or B ($k = 3$) color components begin corrected. The sample $x_{(p,q)2}$ is the (previously corrected) G component located at the center of four surrounding R (or B) components $x_{(p-1,q-1)k}$, $x_{(p-1,q+1)k}$, $x_{(p+1,q-1)k}$, $x_{(p+1,q+1)k}$ forming a square-shape mask (*Fig.2b*). The components $(x_{(i,j)k} - x_{(i,j)2})$ are color-difference quantities. Note that each $x_{(i,j)k}$, for $(i,j) \in \zeta$, is associated with the weight $w_{(i,j)}$ of (2) calculated using original R (or B) components $x_{(p-1,q-1)k}$, $x_{(p-1,q+1)k}$, $x_{(p+1,q-1)k}$, and $x_{(p+1,q+1)k}$ as the inputs.

The correction step (3) updates only third of the interpolated R and B values, therefore, an additional correction step of (3) is needed in order to complete the R and B components $x_{(p,q)k}$ located at positions with original G CFA data. It can be seen that the process forms a spatial arrangement (*Fig.2c,d*) similar to the one used in the G component correction (*Fig.2a*). As before, the G component $x_{(p,q)2}$ is located at the center of four surrounding R (or B) components $x_{(p-1,q)k}$, $x_{(p,q-1)k}$, $x_{(p,q+1)k}$, $x_{(p+1,q)k}$ which now form the diamond-shape mask shown in *Fig.2c,d*. Identical to (1) with $\zeta = \{(p-1, q), (p, q-1), (p, q+1), (p+1, q)\}$, the weighting coefficients $w_{(i,j)}$ are calculated using the R (or B) components $x_{(i,j)k}$, for $(i,j) \in \zeta$, as the inputs.

3. EXPERIMENTAL RESULTS

A number of color images have been used to evaluate the proposed postprocessing framework with representative examples shown in *Fig.3*. All images have been normalized to the standard 512×512, 8-bit per channel RGB representation, except for the Lighthouse image which is 768×512 in size. The tests were performed by sampling the images with the Bayer CFA pattern to obtain a Bayer pattern image [11], demosaicking with a number of different schemes, then applying the proposed method to the restored images. Performance was measured by comparing the original full color images with the demosaicked images obtained with and without the postprocessing/correction step. The mean absolute error (MAE), mean square error (MSE), and normalized color difference (NCD) criterion were used to provide numerical results [16].

The following demosaicking schemes were used for the tests: saturation based adaptive inverse gradient (SAIG) [3], smooth hue transition approach (SHT) [4], median filter interpolation (MFI) [5], adaptive color plane interpolation (API) [6], edge map interpolation (EMI) [7], principle vector method (PVM) [8], color correlation-directional derivatives (C2D2) [9], Kimmel's algorithm (KA) [10], bilinear interpolation (BI) [11], bilinear-difference interpolation (BD) [14], and nearest neighbor interpolation (NNI) [15].

Some visual results in critical image regions are shown in *Figs. 4–7* and the numerical results are summarized in *Tables 1–4*. As can be seen from the numerical results, the proposed method provides modest to significant improvement for almost all cases. In *Fig.4*, the fine structure presents a particular difficulty for demosaicking schemes and results in blurring and various false colors surrounding the structure. After applying the proposed correction step, these effects are significantly reduced. The other images exhibit similar results with the most significant improvement achieved when using the BI and PVM demosaicking schemes.

4. CONCLUSIONS

A demosaicked color image postprocessing technique was presented. Color images restored using a demosaicking scheme are postprocessed/corrected in order to reduce color artifacts and blurring. Edge-sensing weights and color-difference based postpro-

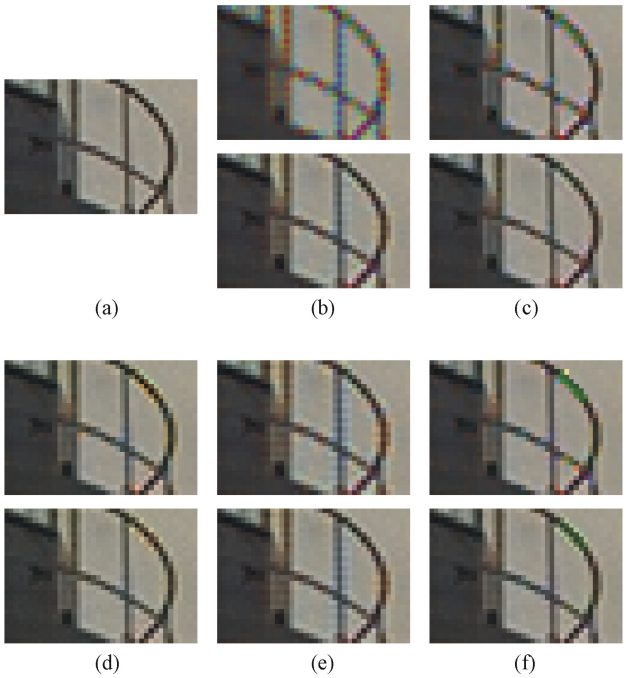

Fig. 4. Enlarged region of the original Lighthouse image (a), and the restored outputs (b)-(f) before (top image) and after (bottom image) postprocessing: (b) BI, (c) PVM, (d) KA, (e) BD, (f) EMI.

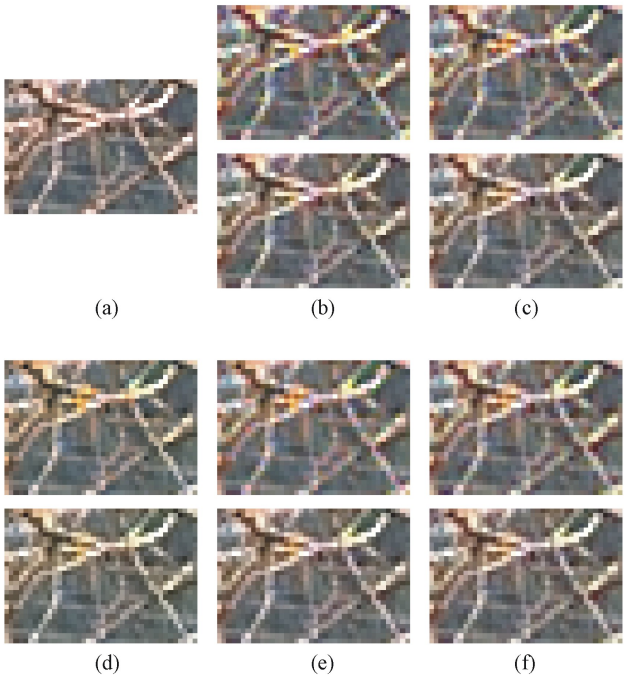

Fig. 6. Enlarged region of the original Train image (a), and the restored outputs (b)-(f) before (top image) and after (bottom image) postprocessing: (b) API, (c) MFI, (d) KA, (e) C2D2, (f) BD.

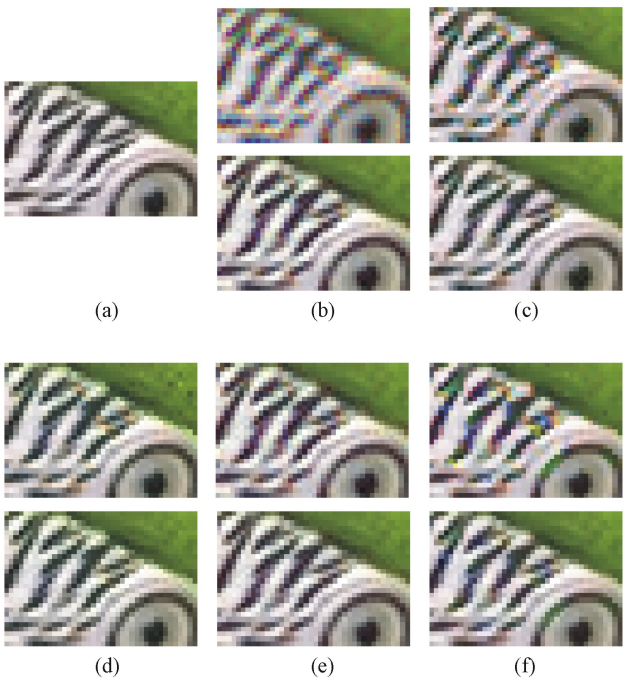

Fig. 5. Enlarged region of the original Parrots image (a), and the restored outputs (b)-(f) before (top image) and after (bottom image) postprocessing: (b) BI, (c) PVM, (d) KA, (e) BD, (f) EMI.

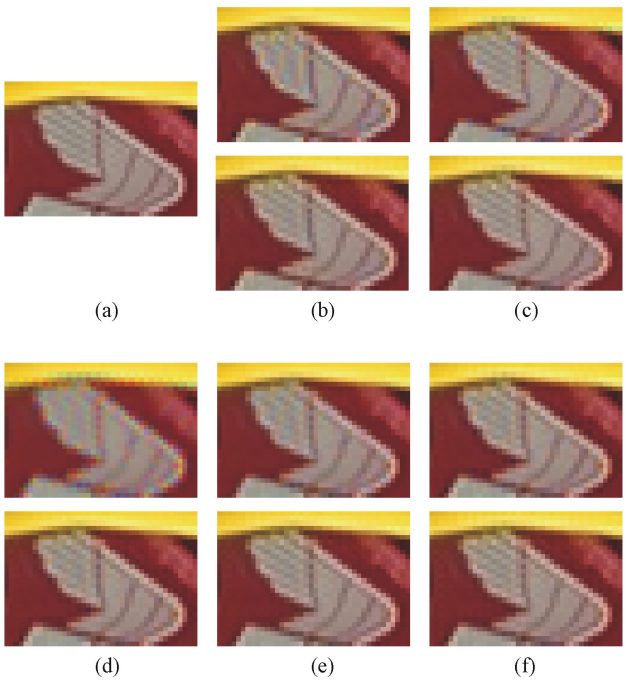

Fig. 7. Enlarged region of the original Bikes image (a), and the restored outputs (b)-(f) before (top image) and after (bottom image) postprocessing: (b) API, (c) MFI, (d) BI, (e) C2D2, (f) BD.

cessing achieve both robust performance and excellent results, in terms of both objective and subjective image quality measures. The proposed method can be implemented as either a correction step incorporated directly into a demosaicking scheme, or as a postprocessing operation performed at a later time. In either scenario, the technique is capable of significantly increasing the quality of color images derived from single-sensor imaging devices.

5. REFERENCES

[1] J. Adams, "Design of practical color filter array interpolation algorithms for digital cameras," *Proc. of the SPIE*, vol. 3028, pp. 117–125, Feb. 1997.

[2] B.E. Bayer, "Color imaging array," U.S. Patent 3 971 065, 1976.

[3] C. Cai, T.H. Yu, and S.K. Mitra, "Saturation-based adaptive inverse gradient interpolation for Bayer pattern images," *IEE Proc. - Vision, Image, Signal Processing*, vol. 148, pp. 202–208, June 2001.

[4] D.R. Cok, "Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal," U.S. Patent 4 642 678, 1987.

[5] W.T. Freeman, "Median filter for reconstructing missing color samples," U.S. Patent 5 373 322, 1988.

[6] J.F. Hamilton and J.E. Adams, "Adaptive color plane interpolation in single sensor color electronic camera," U.S. Patent 5 629 734, 1997.

[7] B.S. Hur and M.G. Kang, "High definition color interpolation scheme for progressive scan CCD image sensor," *IEEE Trans. Cons. Elect.*, vol. 47, pp. 179–186, Feb. 2001.

[8] R. Kakarala and Z. Baharav, "Adaptive demosaicing with the principle vector method," *IEEE Trans. Cons. Elect.*, vol. 48, pp. 932–937, Nov. 2002.

[9] N. Kehtarnavaz, H.J. Oh, and Y. Yoo, "Color filter array interpolation using color correlations and directional derivatives," *J. Electronic Imaging*, vol. 12, pp. 621–632, Oct. 2003.

[10] R. Kimmel, "Demosaicing: image reconstruction from color CCD samples," *IEEE Trans. Image Processing*, vol. 8, no. 9, pp. 1221–1228, Sept. 1999.

[11] P. Longere, Z. Xuemei, P.B. Delahunt, and D.H. Brainard, "Perceptual assessment of demosaicing algorithm performance," *Proc. of the IEEE*, vol. 90, pp. 123–132, Jan. 2002.

[12] R. Lukac, K. Martin, and K.N. Plataniotis, "Colour-difference based demosaicked image postprocessing," *IEE Electronics Letters*, vol. 39, pp. 1805–1806, Dec. 2003.

[13] R. Lukac and K.N. Plataniotis, "A new color restoration solution for Bayer pattern based imaging devices," *IEEE Signal Processing Letters*, to appear, 2004.

[14] S.C. Pei and I.K. Tam, "Effective color interpolation in CCD color filter arrays using signal correlation," *IEEE Trans. Circ. Syst. Video Tech.*, vol. 13, pp. 503–513, June 2003.

[15] T. Sakamoto, C. Nakanishi, T. Hase "Software pixel interpolation for digital still cameras suitable for a 32-bit MCU," *IEEE Trans. Cons. Elect.*, vol. 44, pp. 1342-1352, Nov. 1998.

[16] K. Plataniotis and A. Venetsanopoulos, *Color Image Processing and Applications*, Springer Verlag, 2000.

Table 1. Objective results for the image Lighthouse.

Type	*before postprocessing*			*after postprocessing*		
Method	*MAE*	*MSE*	*NCD*	*MAE*	*MSE*	*NCD*
SAIG	2.570	24.3	0.0403	1.790	9.5	0.0286
SHT	3.619	58.5	0.0522	2.234	20.8	0.0329
MFI	2.341	31.8	0.0380	1.754	13.3	0.0261
API	1.817	12.7	0.0298	1.435	7.0	0.0232
EMI	2.322	23.1	0.0370	1.491	8.3	0.0236
PVM	2.364	23.5	0.0366	1.542	8.6	0.0246
C2D2	2.005	16.2	0.0301	1.483	7.9	0.0231
KA	2.241	19.4	0.0365	1.817	9.6	0.0296
BI	4.468	108.3	0.0653	1.859	13.9	0.0283
BD	2.149	19.7	0.0321	1.725	12.7	0.0262
NNI	6.075	214.5	0.0927	2.337	22.6	0.0354

Table 2. Objective results for the image Parrots.

Type	*before postprocessing*			*after postprocessing*		
Method	*MAE*	*MSE*	*NCD*	*MAE*	*MSE*	*NCD*
SAIG	1.739	12.1	0.0224	1.571	8.0	0.0219
SHT	1.925	18.8	0.0245	1.445	8.3	0.0201
MFI	1.198	7.2	0.0172	1.139	5.1	0.0165
API	1.238	6.2	0.0173	1.200	5.5	0.0168
EMI	1.598	16.9	0.0207	1.259	6.6	0.0176
PVM	1.531	11.8	0.0199	1.187	5.5	0.0166
C2D2	1.216	6.2	0.0163	1.153	5.1	0.0162
KA	2.688	94.6	0.0357	1.991	18.5	0.0270
BI	2.128	31.4	0.0262	1.173	5.6	0.0167
BD	1.202	6.0	0.0170	1.221	6.2	0.0172
NNI	3.317	84.0	0.0430	1.597	11.6	0.0219

Table 3. Objective results for the image Train.

Type	*before postprocessing*			*after postprocessing*		
Method	*MAE*	*MSE*	*NCD*	*MAE*	*MSE*	*NCD*
SAIG	5.985	166.9	0.0866	3.563	53.5	0.0568
SHT	7.759	257.7	0.1085	4.002	65.2	0.0630
MFI	4.873	109.9	0.0798	3.464	51.9	0.0546
API	5.037	117.7	0.0776	3.522	55.5	0.0556
EMI	5.795	170.6	0.0845	3.523	57.5	0.0547
PVM	6.780	214.3	0.0958	3.700	60.5	0.0565
C2D2	4.884	113.0	0.0702	3.148	44.3	0.0493
KA	4.562	91.8	0.0710	3.657	54.5	0.0600
BI	10.569	466.4	0.1450	4.141	73.9	0.0662
BD	4.622	85.8	0.0744	3.458	50.1	0.0571
NNI	14.416	986.6	0.2080	5.124	112.4	0.0828

Table 4. Objective results for the image Bikes.

Type	*before postprocessing*			*after postprocessing*		
Method	*MAE*	*MSE*	*NCD*	*MAE*	*MSE*	*NCD*
SAIG	3.375	53.1	0.0768	2.330	18.4	0.0595
SHT	4.842	94.8	0.1025	2.612	23.3	0.0649
MFI	2.540	27.4	0.0620	1.882	13.5	0.0452
API	2.335	22.6	0.0551	1.915	14.0	0.0467
EMI	3.332	58.8	0.0744	2.177	19.6	0.0529
PVM	3.644	59.0	0.0791	2.155	17.4	0.0521
C2D2	2.481	26.7	0.0545	1.820	12.6	0.0447
KA	3.410	71.7	0.0792	2.821	30.9	0.0720
BI	6.117	157.1	0.1209	2.165	17.4	0.0527
BD	2.396	20.5	0.0581	1.960	14.2	0.0479
NNI	8.903	352.7	0.0188	3.040	33.9	0.0726