Improved Image Accuracy in Hot Pixel Degraded Digital Cameras

Glenn H. Chapman, Rohit Thomas
School of Engineering Science
Simon Fraser University
Burnaby, B.C., Canada, V5A 1S6
glennc@ensc.sfu.ca, rpt4@sfu.ca

Israel Koren, Zahava Koren
Dept. of Electrical and Computer Engineering
University of Massachusetts
Amherst, MA, 01003
koren,zkoren@ecs.umass.edu

***Abstract*— In our previous papers we concluded that the main source of defects in digital cameras are "*Hot Pixels*" and showed that their numbers increase at a nearly constant temporal rate (during the camera's lifetime) and they are randomly distributed spatially (over the camera sensor). The defect characteristics of each hot pixel, i.e., the offset and slope of its *dark response*, appear to remain constant after formation and can, therefore, be extracted and used for image correction. In this paper we suggest a novel method for correcting the damage to the image caused by a hot pixel, based on estimating its dark response parameters. Based on experiments on a camera with 31 known hot pixels, we compare our correction algorithm to the conventional correction done by interpolating the four defective pixel neighbors. We claim that the correction method used should depend on the severity of the hot pixel, on the exposure time, on the ISO, and on the variability of the pixel's neighbors in the specific image we are correcting.**

Keywords- imager defect correction, hot pixel, active pixel sensor APS, CCD, ISO

I. INTRODUCTION

Digital imager technology dominates the world of photography and is becoming ubiquitous, spreading into everyday products from cell phones to cars via embedded sensors. The natural result of this is a push to decrease pixel size and increase imager sensitivity. An important problem is that digital imagers, like other microelectronic devices, develop defects over time, and the nature of the sensor makes it more sensitive to defects that most likely would not affect other devices. Unlike these other devices, most in-field defects in digital imagers begin appearing soon after fabrication, are permanent, and their number increases continuously over the lifetime of the sensor. Clearly, the quality of the image generated by the sensor is degrading.

Over several years now we have been investigating imager in-field defect development and have identified the characteristics and rate of faulty pixels [1-6]. Furthermore, based on the random locations of defective pixels, we have identified the in-field defect causal mechanism as most likely being cosmic ray damage, which cannot be protected against by methods such as shielding. More importantly, our recent studies resulted in an empirical formula, which projects that as the pixel size shrinks, and the sensitivity increases, defect numbers will grow via a power law of pixel size to the -3.3. This formula predicts that as pixel sizes drop below two microns, and sensitivities move towards those for low light night pictures, defect rates can grow to hundreds or even thousands per year in typical cameras. At these rates, the conventional correction methods based on simple averaging of the faulty pixel's neighbors may not be the best, as one (or more) of the neighbors may be faulty as well. This paper suggests a novel algorithm for correcting images based on the knowledge of the pixel defect parameters. We devise a method for estimating the true value of the defective pixel, and based on it we experimentally compare the improvement in pixel defect correction achieved using our algorithm to the improvement using the conventional method of replacing the pixel by an interpolation of its four neighbors.

This paper is organized as follows: Section II describes the most common type of imager defects we encountered, namely *hot pixels*. Section III describes the growth rate of the hot pixels. Section IV presents the algorithm we suggest to correct these defective pixels. Section V describes the numerical experiments we conducted to validate the effectiveness of our algorithm, and Section VI concludes the paper.

II. HOT PIXELS

Over the past 8 years [5,6], we manually calibrated many commercial cameras, including 24 Digital Single Lens Reflex (DSLRs), using dark field exposure (i.e., no illumination) to identify stuck-high and partially stuck defects. However, we have not found any of these stuck defect types, even though they are commonly discussed in camera forums. Instead, hot pixels (i.e., a pixel that appears as a bright dot in every image) were the dominant defect type. In the past, researchers believed that a hot pixel has an *illumination-independent* component that increases linearly with exposure time. Our experiments showed that this component does in fact depend on the illumination, and has, therefore, to be identified by capturing a series of images both at increasing exposure times and at increasing illumination. Figure 1 demonstrates the *dark response* of a hot pixel by showing the normalized pixel illumination vs. the exposure time where illumination level 0 represents no illumination and level 1 represents saturation. The dark response of a good pixel should be close to 0 (with some variation due to noise in the sensor) at any exposure level, as shown by curve (a) in the figure. In addition, we have found [5] that hot pixels can be categorized into two types: standard hot pixels (see curve (b) in Figure 1), which have a component that increases linearly with exposure time;

978-1-4799-1585-9/13/$31.00 ©2013 IEEE

and partially stuck hot pixels (curve (c)) which have an additional offset that can be observed at zero exposure.

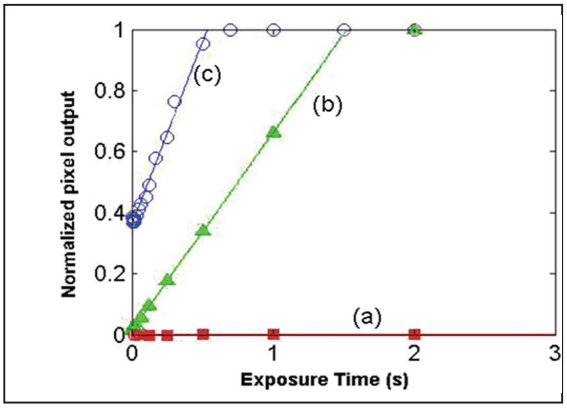

Figure 1: Comparing the dark response of imager pixels (a) good pixel, (b) standard hot pixel, (c) hot pixel with offset.

While the overall imager system is digital, the sensor is an analog device. The response ***I*** of both a good and hot pixel can be modeled using Equation (1),

$$I(R,T) = a + bT + cRT \quad (1)$$

where R measures the incident illumination rate, T measures the exposure time, and a,b are the hot-pixel parameters, which are a function of the ISO (sensitivity). In the ideal hot pixel c =1 for a given ISO.

If a *dark-field image* is taken, then $R=0$ and the response of a hot pixel becomes

$$I_{hot} = I(0,T) = a + bT \quad (2)$$

For a good pixel, $a=b=0$ and

$$I(R,T) = cRT \quad (3)$$

The expression for the dark response (also called the combined dark offset), shown in Equation (2), is linear. Thus, by plotting the pixel dark response vs. exposure time, as shown in Figure 1, a linear function can be used to estimate a and b. For a standard hot pixel, a is zero, and therefore this type of defect is most visible in long exposure images. In contrast, for partially stuck hot pixels, the response depends on the magnitude of a, and this type of defect will appear in all images.

To obtain this data for a given camera involves typically 5 to 20 calibration images per test at a wide range of exposure times and ISOs, and their analysis with specialized software [2-4]. Still, this kind of analysis is relatively easy to do, and the *dark-field* parameters can be readily obtained.

In our long duration study we have identified hot pixels from 24 DSLR cameras including both APS and CCD sensors, with the age of these cameras ranging between 1 and 7 years [9]. After performing the dark-frame calibration at ISO 400, our results showed a cumulative total of 243 hot pixels of which 44% were of the partially stuck type. The offset in the partially stuck hot pixels causes this type of defect to appear at any exposure level, and thus has a greater impact on the image quality. The imager's ISO setting controls the sensitivity or amplification of the pixel output. Higher ISO setting enables objects to be captured under low light conditions or with very short exposures. This allows natural light photography without the need for flash or a long exposure time. The amplification level scales proportionally with the ISO setting, but the usable ISO range is limited by the noise level of the sensor. Ten years ago, most commercial DSLRs had a usable ISO range of 100 to 1600. As sensor technology improved, noise levels have been reduced and the usable ISO range has increased considerably, with recent DSLRs having an ISO range of 50 to 12,300 and high-end cameras having a range from 25,600 to 409,600 ISO. The higher ISOs increase significantly the visibility of the hot pixels.

The high number of hot pixels with offsets suggests that the development of stuck high pixels in the field may actually be due to the presence of hot pixels with very high offsets. This is consistent with our experience of not having detected a true stuck pixel in our cameras, while explaining the cameras developing stuck pixels discussed in camera forums.

III. DEFECT GROWTH RATE

In our previous research we have shown that hot pixel defects occur randomly spaced across the imager [1-6]. This indicates they are created by a random source, such as cosmic rays. Other authors have seen the same result, and have shown that neutrons seem to create the same hot pixel defect types [7,8]. Most recently, in [9], we developed an empirical formula relating the defect density D (defects per year per mm^2 of sensor area) to the pixel size S (in microns) and sensor gain (ISO) via the following equations:

For APS pixels

$$D=10^{-1.13}\, S^{-3.05} ISO^{0.505} \quad (4)$$

and for CCD sensors

$$D=10^{-1.849}\, S^{-2.25} ISO^{0.687} \quad (5)$$

The importance of these equations is that they show that the defect rate rises rapidly when the pixel size goes below 2 microns (see Figure 2), and is projected to reach 12.5 defects/year/mm^2 at ISO 25,600 (which is already available on some high-end cameras).

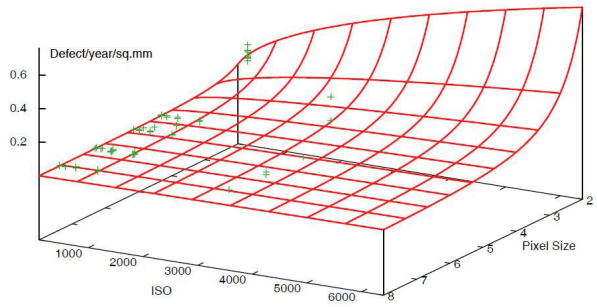

Figure 2: Fitted power law for APS defect density (defects/year/mm^2) vs pixel size (μm) and ISO

Since current camera trends are to drastically reduce pixel sizes, our experimental results project that the number of hot pixel defects will increase to such a level where correcting the image will be crucial.

IV. Model and Algorithm For Defect Correction

A digital imager can be modeled as an array of $U \times V$ pixels, with x_{ij} denoting the incident illumination at location (i,j) for a given image. Every x_{ij} consists of separate pixel values, each pertaining to a different color component. Most color imagers use a repeating tile pattern of the Bayer Color Filter Array (CFA) [3] (red, blue, and two greens – see Figure 3), so that the red, blue, and two green channels can be treated independently. For the purpose of this analysis we will define a repeated CFA pattern as a single CFA pixel. When camera data is extracted, the individual colors form single pixels, 4 of which make up this CFA pixel.

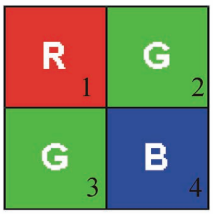

Figure 3: Bayer Color Filter Array with k numbering

Denoting by $x_{ij}^{(k)}$ the incident illumination of color k (k=1,2,3,4 – see Figure 3), we standardize it so that $0 \le x_{ij}^{(k)} \le 1$.

We then denote by $y_{ij}^{(k)}$ the (standardized) sensor reading of color k in location $(i.j)$ (i=1,…,U ; j=1,…,V ; k=1,2,3,4). For a defect-free CFA pixel, $y_{ij}^{(k)} = x_{ij}^{(k)}$ *for all k=1,...,4.*

Since the hot pixel defects are very small, at most one of the color components per CFA pixel will be hot, and for this k

$$y_{ij}^{(k)} = x_{ij}^{(k)} + a + bT \qquad (6)$$

where $a+bT$ is the offset due to the hot pixel defect.

For simplicity, we remove in the following discussion the indices i,j,k and number the hot (color) pixels m=1,...,M (where M is the number of hot pixels). x_m denotes the illumination and y_m the sensor reading of hot (color) pixel m.

Figure 4 shows the defective pixel in the center with the surrounding neighbor pixels. Any of the R,G,G,B in the center can be hot.

For our correction algorithm we use the following notations

$A_m^{(4)}$ = Conventional corrected value of hot pixel m based on 4 neighbors = Average of 4 nearest neighbors

For example, if the color Red at i,j is faulty, then this averages the values of R (or k=1) for $x_{i-1,j}$, $x_{i+1,j}$,$x_{i,j+1}$, $x_{i,j-1}$

$A_m^{(8)}$ = Conventional corrected value of hot pixel m based on 8 neighbors = Average of 8 nearest neighbors

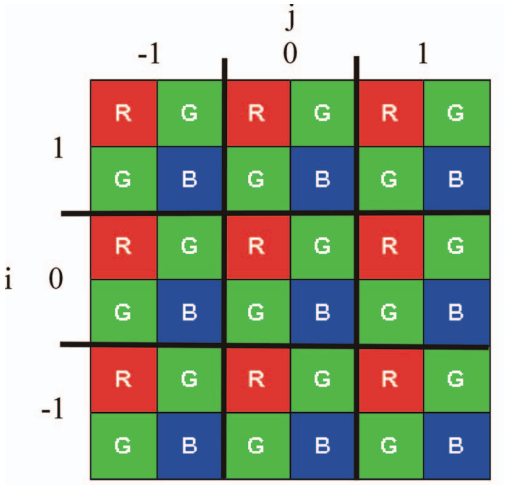

Figure 4: Pixel color array showing surrounding pixels with relative i,j

Again, for the color Red (k=1) example this averages

$x_{i-1,j-1}$, $x_{i,j-1}$,$x_{i+1,j-1}$, $x_{i-1,j}$, $x_{i+1,j}$, $x_{i-1,j+1}$, $x_{i,j+1}$,$x_{i+1,j+1}$

We now denote by D_m a partially-corrected value based on the *dark response* parameters of the hot pixel (recall that these are relatively easy to obtain)

$$D_m = y_m - (a + b\,T) \qquad (7)$$

Note that the 4 and 8 point interpolations only give good results when the 9 pixels of Figure 4 have a light that changes slowly across the image for the given color. This is effectively a tilted plain of that color. In practice, these interpolations fail badly where an edge or sudden change occurs anywhere in that 9-pixel set. This constitutes quite a large area of the camera image, so such changes often occur. Hence, if we can correct the image using the hot pixel parameters in those rapidly changing locations, we may get a better image correction. Still, our corrected value D_m (Equation 7) may not be totally accurate either as it is based on curve fitting and only on darkfield measurements. We therefore suggest the following correction algorithm which uses a weighted combination, denoted by C_m , of A_m and D_m .

Our algorithm differentiates between uniform areas on the image and rapidly changing areas by comparing the two averages $A_m^{(4)}$ and $A_m^{(8)}$ - if they differ by less than a threshold ε, the area is considered uniform, otherwise it is considered "busy". We allow two different sets of weights,

α ,(1 - α) and β ,(1 - β) depending on whether the neighborhood is uniform or busy, respectively.

Weighted_Correction_Algorithm (8)

For a hot-pixel value y_m
Select $\varepsilon \geq 0,\ 0 \leq \alpha \leq 1,\ 0 \leq \beta \leq 1$

If $abs(A_m^{(4)} - A_m^{(8)}) \leq \varepsilon$ (indicating a slowly changing area)
replace y_m by $C_m = \alpha A_m^{(4)} + (1 - \alpha) D_m$

Otherwise (indicating sudden changes)
replace y_m by $C_m = \beta A_m^{(4)} + (1 - \beta) D_m$

If $y_m \geq 0.99$ (indicating saturation)
replace y_m by $C_m = A_m^{(4)}$

The algorithm parameters ε, α, and β need to be selected empirically.

V. Experimental Measurements

To test our algorithm we used the oldest (8 years) DSLR we have been testing for the last 6 years. This camera has 31 hot pixels of varying strengths at the ISO 800 level, which is the highest gain a camera of this era could get without noise starting to dominate the image. We also needed a busy scene, similarly to those of typical images. Using an image that is nearly uniform (say a uniformly illuminated gray wall) would not test our algorithm as for such a scene the interpolation would always give perfect results, yet this is not a typical picture taken by photographers.

We chose as a test pattern a busy scene, specifically a wall of books, so that the image changes in many places, but all objects are at about the same distance from the camera (see Figure 5). This scene has some areas that are slowly changing, good for the interpolation methods, and other areas that are rapidly changing, where the D_m should perform better.

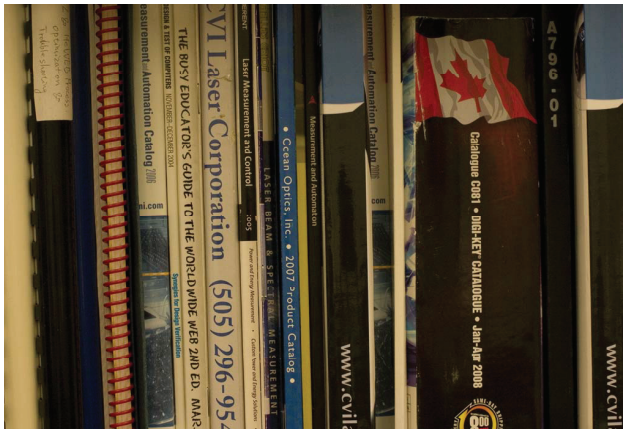

Figure 5: Test image for pixel correction

Note that the exposure for the scene was selected so that no picture areas were saturated (i.e., the pixel is at maximum value where it no longer responds to changes in illumination or to the effect of the hot pixel). At the longest exposure some of the hot pixels did create saturation, and those will not be included in this test.

The problem in experimentally testing our algorithm is that we need to compare its output to the *real value* for the defective pixel at the exact location, which clearly is not easy to obtain. We take advantage of equation (2) which notes that as the exposure time T is decreased to very short values, the hot pixel contribution $I(R,T)$ is very small. This exposure time change would, if uncompensated for, reduce the pixel light intensity. However, we want to keep the picture collected illumination, and thus each pixel's collected light RT, constant, so only the hot pixel effect is being reduced. We can compensate by changing the optical system to reduce the light intensity so that R/T is a constant R_1: e.g., if $R=R_1$ for $T=1$ sec, then at $T=0.008$ sec $R=R_1/0.008 = 125R_1$. Technically this is done by changing the lens aperture of the camera (F#) from F=.22 to F=2. Using this semiprofessional DSLR with a fast lens allowed us to take scenes with exposure times T reduced in steps of a factor of two from 1 second to 0.008 second. All the pictures in these tests will thus have the same RT value for a given pixel in the scene, though the value of RT varies across Figure 5 between different pixels.

To find the value of X (i.e., the *true value* of the pixel) for each of the hot pixels in the image of Figure 5, we need to subtract the hot pixel contribution. For the shortest exposure, the hot pixel contribution is the smallest but still needs to be compensated for.

The effect of light on hot pixels is not discussed in the literature. To find this impact we used our test camera and took a series of images with uniform illumination (R), but varying exposure times from 1 sec to 0.001 sec, where there should be almost no contribution from the cRT term (in Equation 1). This was done for a series of illuminations ranging from $R=0$ (dark field) to RT nearly saturated. For these test images, we could extract for the hot pixels the measured values Y. Moreover, as the image had almost perfectly uniform illumination, we could extract the true values X at each hot pixel by the simple 4-neighbor interpolation $A_m^{(4)}$ for each exposure time T. Using 44 triplets (X,Y,T) for each hot pixel, we calculated the best fitted curves

$$Y = a + bT + cX$$

These resulted in slightly changed a and b parameters compared to the dark field parameters calculated previously. The parameter c was found to remain nearly 1. Once we obtained a,b,c for each hot pixel, we could go back to the 0.008 second image of Figure 5 and correct each of the hot pixels using

$$X = (Y - a - bT)/c \qquad (9)$$

We defined this as the *true value* X of the pixel at the given location. This would be the reading if there was no defect there.

Since the exposure time T is changed but RT is constant in all the test pictures, the only changes could be due to the hot pixel. Hence X will remain the same for all the other exposures allowing us to test our correction algorithm for the hot pixels for a variety of hot pixel parameters.

Table 1: Hot pixel test data for 1/30th second on test image showing the results for the correction algorithms

HotPix#	Y_m measured	X_m True	$I(0,0.03)$	D_m	$A_m^{(4)}$	$A_m^{(8)}$	C_m	C_m error – $A_m^{(4)}$ error
1	0.2228	0.1546	0.0425	0.1803	0.1182	0.1226	0.1524	-0.0342
2	0.1295	0.0756	0.0412	0.0883	0.0444	0.0441	0.0760	-0.0308
3	0.0767	0.0271	0.0147	0.0619	0.0388	0.0319	0.0515	0.0127
4	0.0498	0.0346	0.0159	0.0339	0.0284	0.0351	0.0314	-0.0031
5	0.1633	0.0647	0.0895	0.0738	0.0348	0.0351	0.0629	-0.0281
6	0.2353	0.1981	0.0317	0.2036	0.1510	0.1491	0.1889	-0.0379
7	0.3578	0.1408	0.0583	0.2994	0.1490	0.1455	0.2573	0.1083
8	0.1015	0.0562	0.0267	0.0747	0.0637	0.0635	0.0716	0.0079
9	0.0967	0.1243	0.0119	0.0848	0.1005	0.1172	0.0919	0.0086
10	0.1412	0.0586	0.0405	0.1007	0.0345	0.0335	0.0822	-0.0004
11	0.3497	0.0465	0.3098	0.0398	0.0448	0.0483	0.0412	0.0036
12	0.0821	0.0466	0.0270	0.0552	0.0167	0.0190	0.0444	-0.0277
13	0.2658	0.0848	0.1657	0.1001	0.0106	0.0120	0.0750	-0.0644
14	0.0322	0.0158	0.0107	0.0216	0.0049	0.0093	0.0169	-0.0098
15	0.0547	0.0533	0.0057	0.0490	0.0335	0.0348	0.0447	-0.0112
16	0.0671	0.0439	0.0186	0.0485	0.0212	0.0207	0.0409	-0.0197
17	0.3004	0.1135	0.1815	0.1189	0.0222	0.0241	0.0918	-0.0696
18	0.9218	0.3601	0.4131	0.5088	0.0051	0.0045	0.3677	-0.3473
19	0.2939	0.1143	0.1142	0.1797	0.0358	0.0272	0.1150	-0.0778
20	0.0749	0.0285	0.0272	0.0477	0.0144	0.0141	0.0383	-0.0041
21	0.0511	0.0217	0.0126	0.0385	0.0108	0.0113	0.0307	-0.0020
22	0.0136	0.0038	0.0190	0.0010	0.0077	0.0080	0.0029	-0.0030
23	0.4590	0.1658	0.2756	0.1834	0.0214	0.0224	0.1380	-0.1166
24	0.0735	0.0258	0.0325	0.0410	0.0051	0.0058	0.0310	-0.0155
25	0.0723	0.0325	0.0304	0.0419	0.0057	0.0053	0.0317	-0.0260
26	0.1802	0.1206	0.0854	0.0948	0.0197	0.0193	0.0738	-0.0540
27	0.1653	0.1109	0.0806	0.0846	0.0207	0.0206	0.0667	-0.0460
28	0.0372	0.0156	0.0096	0.0276	0.0134	0.0116	0.0236	0.0057
29	0.1986	0.1664	0.0010	0.1976	0.1835	0.1847	0.1937	0.0101
30	0.0050	0.0075	0.0001	0.0049	0.0068	0.0066	0.0054	0.0014
31	0.0065	0.0048	0.0006	0.0059	0.0075	0.0076	0.0064	-0.0011

For each of the longer exposure time tests of this image we extract the measured value Y_m at the hot pixel, and compare it to the true value. Each exposure step is a doubling of the hot pixel dark current effect, and rapidly the dark current dominates the pixel value.

Table 1 shows the data set for 1/30th second exposure time, a typical value for low light images. All the values are scaled so 1 would be the maximum value of the pixel. With the hot pixel data obtained from dark field measurements (see Section II) we calculated for each hot pixel the offset $I(0,1/30)$.

We then calculated the different correction functions D_m, $A_m^{(4)}$ and $A_m^{(8)}$, and finally our proposed weighted correction C_m (as described in 8). The best results for our algorithm were found by selecting as its parameters: ε= 0.0055 (the threshold distance between the averages of 4 and of 8 neighbors under which we call the area uniform), α=0.45 (the weight of interpolation for uniform areas), β=0.28 (the weight of interpolation for rapidly changing areas) . Note that in both cases the dark field correction result has the higher (>0.5) weight compared to interpolation (0.55 and 0.72).

We then compared (by subtraction) the interpolation errors $abs(X_m - A_m)$ and the errors of our algorithm $abs(X_m - C_m)$ in the last column of Table 1. A negative number there means the our algorithm gives a better result than the 4-neighbor average. The results for the scene in Figure 5 at 1/30th sec exposure time show that in 23 cases out of 31, or 71%, our algorithm performed better. More importantly, the average absolute error for C_m is 0.015, compared to an average error of 0.043 for the 4-neighbor interpolation $A_m^{(4)}$ (which is the currently used commercial method) and an average error of 0.026 for the dark field-based correction D_m.

This clearly shows the advantage of our algorithm, which takes into account the hot pixel parameters and the variability of the local pixel area, over simple interpolation techniques.

VI. Conclusions

This paper has presented a novel method for improving the images taken by a camera with hot pixels, by using the hot pixel's dark response parameters in combination with the conventional nearest neighbors averaging. In order to compare our method to the conventional interpolation method, we needed to devise a new procedure for extracting the correct pixel value and then calculate the error introduced by the different methods. We conclude that we obtained significantly better results using our weighted correction method, which gives greater weight to the hot pixel correction when the pixel area is rapidly changing, and a slightly smaller weight (although still >0.5) in more uniform areas. Future work will extend these results to include cameras with higher ISO ranges.

References

[1] J. Dudas, L.M. Wu, C. Jung, G.H. Chapman, Z. Koren, and I. Koren, "Identification of in-field defect development in digital image sensors," *Proc. Electronic Imaging, Digital Photography III*, v6502, 65020Y1-0Y12, San Jose, Jan 2007.

[2] J. Leung, G.H. Chapman, I. Koren, and Z. Koren, "Statistical Identification and Analysis of Defect Development in Digital Imagers," *Proc. SPIE Electronic Imaging, Digital Photography V*, v7250, 742903-1 – 03-12, San Jose, Jan 2009.

[3] J. Leung, G. Chapman, I. Koren, and Z. Koren, "Automatic Detection of In-field Defect Growth in Image Sensors," *Proc. of the 2008 IEEE Intern. Symposium on Defect and Fault Tolerance in VLSI Systems*, 220-228, Boston, MA, Oct. 2008.

[4] J. Leung, G. H. Chapman, I. Koren, Z. Koren, "Tradeoffs in imager design with respect to pixel defect rates," *Proc. of the 2010 Intern. Symposium on Defect and Fault Tolerance in VLSI*, 231-239., Kyoto, Japan, Oct 2010.

[5] J. Leung, J. Dudas, G. H. Chapman, I. Koren, Z. Koren, "Quantitative Analysis of In-Field Defects in Image Sensor Arrays," *Proc. of the 2007 Intern. Symposium on Defect and Fault Tolerance in VLSI*, 526-534, Rome, Italy, Sept 2007.

[6] J. Leung, G.H. Chapman, Y.H. Choi, R. Thomson, I. Koren, and Z. Koren, "Tradeoffs in imager design parameters for sensor reliability," Proc*., Electronic Imaging, Sensors, Cameras, and Systems for Industrial/Scientific Applications XI*, v 7875, 78750I1-0I12, San Jose, Jan. 2011.

[7] A.J.P. Theuwissen, "Influence of terrestrial cosmic rays on the reliability of CCD image sensors. Part 1: experiments at room temperature," *IEEE Transactions on Electron Devices*, Vol. 54 (12), 3260-6, 2007.

[8] A.J.P. Theuwissen, "Influence of terrestrial cosmic rays on the reliability of CCD image sensors. Part 2: experiments at elevated temperature," *IEEE Transactions on Electron Devices*, Vol. 55 (9), 2324-8, 2008.

[9] G.H. Chapman, R. Thomas, I. Koren, and Z. Koren, "Empirical formula for rates of hot pixel defects based on pixel size, sensor area and ISO", *Proc. Electronic Imaging, Sensors, Cameras, and Systems for Industrial/Scientific Applications XIII*, v8659, 86590C-1-C-11 San Francisco, Jan. 2013.