High-dynamic range tone-mapping algorithm for focal plane processors

S. Vargas-Sierra, G. Liñán-Cembrano, E. Roca, A. Rodríguez-Vázquez
Instituto de Microelectrónica de Sevilla (IMSE-CNM), CSIC and Universidad de Sevilla
41092 Sevilla, Spain

ABSTRACT

This paper presents a Dynamic Range improvement technique which is specially well-suited to be implemented in Focal Plane Processors (FPP) due to its very limited computing requirements since only local memories, little digital control and a comparator are required at the pixel level. The presented algorithm employs measurements during exposure time to create a 4-bit non-linear image whose histogram determines the shape of the tone-mapping curve which is applied to create the final image. Simulations results over a highly bimodal 120dB image are presented showing that both the highly and poorly illuminated parts of the image keep a sufficient level of details.

Keywords: High Dynamic Range, CMOS Image Sensor, Tone Mapping

1. INTRODUCTION

High Dynamic Range (HDR)[1] vision is required in several applications like automotive –both for indoor and outdoor monitoring–, surveillance, scientific experiments, etc. CMOS technologies allow for including processing circuitry very close to the sensors, in the well-known focal-plane approach, giving us the possibility to process the raw information provided by photosensors through a myriad of dynamic range expansion algorithms[1],[2] or even, to dynamically adapt the response of the sensors –i.e. the sensor's transduction function– according to the existing lighting conditions.[3]

Almost all imagers reported –and commercialized– codify absorbed photons in terms of a voltage –or a voltage difference to be more precise– instead of directly using the photogenerated current –or carriers. Typically, photogenerated current is integrated for a given time –exposure– in a previously precharged capacitor, producing a quasi-linear relationship[*] between incremental voltage and photogenerated carriers, and consequently, incident photons. The most common alternative to this linear approach consists of forcing the photogenerated current to flow through a compressive-type current-to-voltage device (by definition: a non-linear resistor) to produce a voltage which is a compressed version of the photogenerated current. Logarithmic sensors,[4] where the non-linear resistor's role is played by a MOS transistor in weak inversion, are the most remarkable example of this sensing mechanism. One of the mjor drawbacks of this approach is the use of a fixed compression scheme for all possible illumination scenarios.

Regarding the non-compressive approach, working with linearly captured HDR images requires employing long bit-words per pixel, which are more difficult to handle by either processors or visualization devices than the typical 8-bit coding per color channel. Tone Mapping (TM) techniques[5] can be applied to transform the colors (or intensities) of HDR images into a lower bit count representation in order to make the HDR images directly usable by conventional processors or displays. Obviously, here the goal is to minimize the loss of visual information –or to maximize image quality, which, as in our work, does not necessarily mean the same thing– after word-length reduction.

The tone mapping technique reported in this work has been specially designed for a full implementation within a Focal Plane Processor (FPP),[6] thus it is both computationally simple and requires neither intensive memory accesses nor memorization of many images for long time. The algorithm produces 7-bit images –that could be stored at the pixel level in SRAM units– through a mechanism which dynamically adapts to the contents on the scene. Shortly, if we only have 2^7 codes available and the DR to cover is much wider, we need to assign output codes non-uniformly, more codes to some ranges of illuminations and less –or none– to other ranges. In our work, this decision is made upon the supposition that a down-scaled[†] (×1/4) version of the previous image is a good indicator of the probability of a range of illuminations to be significant in the current image. Besides, since the algorithm is to be fully integrated in a FPP, only required calculations are accumulations, comparisons and divisions by integer constants.

Further author information:
S.V-S.: Email: sonia@imse-cnm.csic.es, Telephone: +34 954466634

[*] Due to capacitor non-linearities.

[†] In order to reduce computational cost.

Bioelectronics, Biomedical, and Bioinspired Systems V; and Nanotechnology V, edited by Ángel B. Rodríguez-Vázquez,
Ricardo A. Carmona-Galán, Gustavo Liñán-Cembrano, Rainer Adelung, Carsten Ronning, Proc. of SPIE Vol. 8068,
806807 · © 2011 SPIE · CCC code: 0277-786X/11/$18 · doi: 10.1117/12.889448

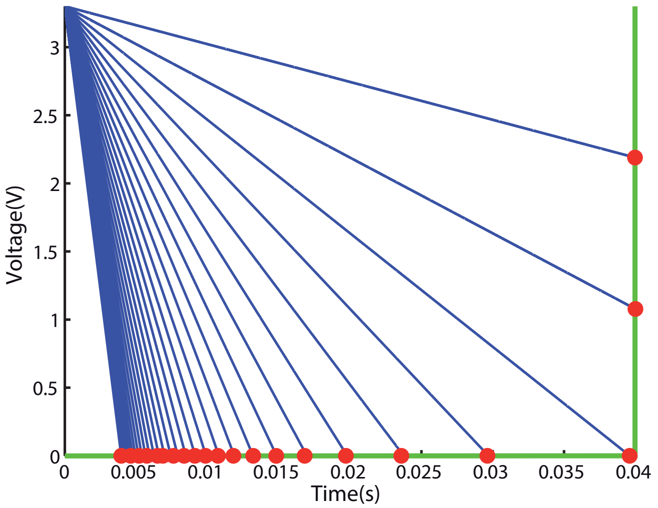


Figure 1. Intersections between equally distributed I_{ph} discharges and ideal voltage reference.

2. LIGHT INTENSITIES SAMPLING METHOD

The presented Tone Mapping Algorithm is fully oriented to its hardware implementation in integration pixels. In these pixels, the photocurrent, I_{ph}, discharges a capacitor, C_{ph}, which has been previously initialized to V_{rst}. If one measures the voltage drop across this capacitor, V_{pixel}, it evolves as (1). Conversely, one could also measure the time (T_{cross}) which takes V_{pixel} to reach a reference voltage V_{ref}, obtaining (2). Easily, one observes that using the same circuitry and operation principle, we obtain a voltage that is directly proportional to the photogenerated current in the first case, and a time which is inversely proportional to it in the second case.

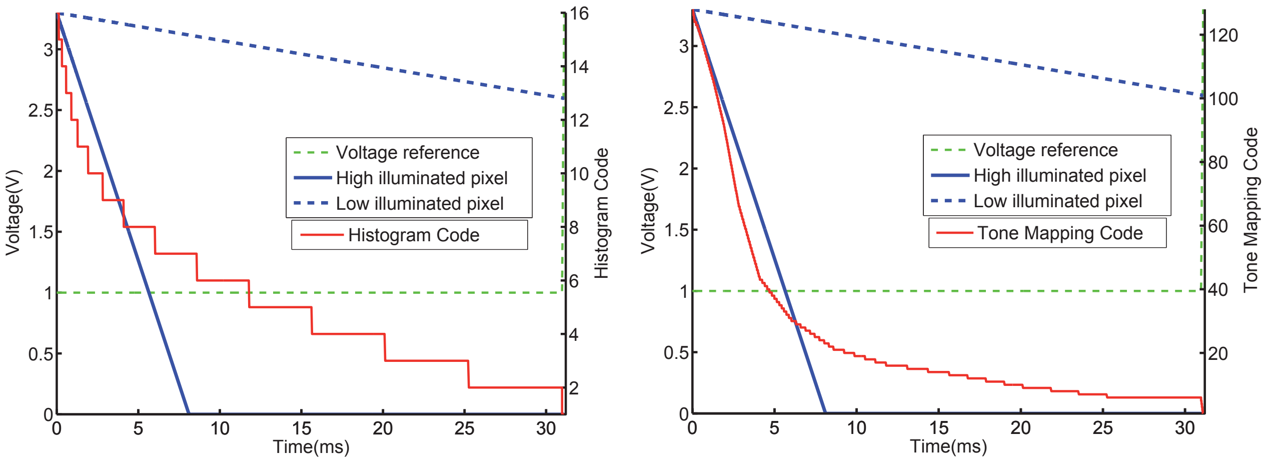
$$V_{pixel}(I_{ph}, t) = V_{rst} - \frac{I_{ph}}{C_{ph}} \cdot \Delta t \tag{1}$$

$$T_{cross}(I_{ph}, V_{ref}) = \frac{C_{ph}}{I_{ph}}(V_{rst} - V_{ref}) \tag{2}$$

Let us illustrate this using Fig. 1. Here, red dots show the intersection between the discharge signals V_{pixel} (in blue) of uniformly distributed photocurrents with a fixed voltage reference (green line). The reset voltage is 3.3V and maximum exposure time is 40ms. If we measure V_{pixel} values at the end of the exposure, we see that uniformly distributed currents produce uniformly distributed pixel voltages when not saturated. However, if we check the times required to saturate the sensor for linearly increasing photocurrents (to reach 0V in this example), we see that times are gradually closer to each other despite the photocurrent difference is the same. These complementary behaviors are the basis of our algorithm for DR expansion. The proposed sampling method will be either measuring the time of intersection of the discharge signals with the voltage reference for the cases this situation happens within the exposure window, or measuring the final voltage for the cases it does not. This natural compression derived by the sampling method will be tone-mapped afterwards by the algorithm assignation of digital output codes.

3. TIME-VOLTAGE HISTOGRAM IMAGE

Our HDR algorithm combines these two kind of measurements, voltage measurements at the end of the exposure and crossing-time measurements during exposure. The aim is to create an artificial image whose histogram can be used to determine the distribution of light intensities in the scene. We call this artificial image the Histogram Image (HI). For FPP implementation purposes, we will limit the final image (denoted as Tone-Mapped Image –TMI) to 7-bit and the HI to just



(*a*) *Histogram Image*

(*b*) *Tone Mapped Image*

Figure 2. Images Capture.

4-bit –meaning that only 16 bins are considered–, besides this HI is not full-resolution but a $\times 1/4$ down-scaled version of the resolution of TMI (QCIF[‡] in the final silicon implementation).

Fig. 2(a) shows an example of the generation of HI for two different illuminations. As we see, we compare the pixel voltage with a fixed reference (1V in this case) and we store locally the value of the 4-bit histogram curve at the crossing point. At the end of the 15^{th} period, we ramp-up the reference signal V_{ref} very fast (from 1V to 3.3V in 128µs) to ensure that pixels not crossing previously are assigned the value 1. Obviously, the key of this part of the algorithm is the duration of each code in the histogram curve. Codes are non-linearly distributed in time, in a similar way as they are the crossing times for uniformly distributed illuminations.

During every bin period, $2^{7\ bits} = 128$ evaluations are executed in order to check whether the pixel signals has crossed the voltage reference. The time between evaluations in each bin will determine the bin time. Table 1 shows the *Time per bin* for 1µs evaluation clock period.

The histogram of HI will be employed to determine how the 128 available codes (7-bit image) will be distributed during the application of the tone-mapping that creates the TMI.

4. TONE-MAPPED IMAGE

TMI is generated as follows (see Fig. 2(b)). First of all, we have to say that we could either repeat the exposure, for still scenes, or use the previous HI as an indicator of probability rather than an exact evaluation of the distribution of light intensities within the current scene. The operation principle is the same. We store the value of a 7-bit signal (TMI curve) when V_{pixel} crosses V_{ref}. The TMI curve is generated from the cumulative histogram of HI. Once we know how many

[‡]144x176 pixels.

Table 1. Example of bins distribution.

Bin index	Clocks per evaluation(µs)	Time per bin (µs)	Bin index	Clocks per evaluation(µs)	Time per bin (µs)
16	1	128	8	15	1920
15	1,5	192	7	20	2560
14	2	256	6	25	3200
13	2,5	320	5	30	3840
12	3	384	4	35	4480
11	5	640	3	40	5120
10	7	896	2	45	5760
9	10	1280	1	1	128

pixels crossed the reference voltage within the different time windows (bins), we can calculate the relative contribution of this bin to the final image –number of pixels in this bin over total number of pixels –, and assign output codes within a bin accordingly to the relevance of this bin in HI. This operation enhances (by assigning more output codes) the most important light intensity ranges in the image at the expense of the not relevant ones.

The TMI curve will be generated decreasing[§] one code any time the data retrieved from a Look Up Table(LUT) indicates it. Each of the 129 position of this LUT(0-128 levels per bin) contains a 128-bit word which denotes the positions of code changes in the 128 evaluation cycles depending of the assigned levels per bin. It is worth to mention that within a bin, codes are uniformly distributed in time –according to the duration of that bin. Code changing positions are calculated by the computation of $\frac{i \cdot 128}{N}$ where N is the levels per bin and $i = 1 - N$. These positions are then flipped left-to-right to obtain a code decrement in the first position of all levels except for position 0, which corresponds to the case that no codes assigned to a bin, this allows to distinguish the codes in any new bin. Moreover, the first code changing order needs to be discarded as it is not necessary.

5. ALGORITHM

In order to execute this HDR technique one must go through these steps:

1. Define the duration of the bins –that is, the location of our 16 temporal windows.
2. Set default image levels per bin for the first image → a priori all bins could have the same number of codes, indeed this first image can be discarded as the role for the first frame is the creation of HI.
3. Capture Histogram Image.
4. Compute the Histogram.
5. Calculate levels per bin to be assigned.
6. Compute the tone mapping curve → evaluate, according to the codes per bin and duration of each bin, when to decrease the tone-mapping curve.
7. Produce TMI.

As it has been already mentioned, we can take the previous image as an indicator of probabilities of light intensity distribution within the current scene, besides, since the generation of HI and TMI requires the same waveform for V_{ref}, we can simultaneously create TMI for this frame and HI for the next one. At the beginning of every exposure, the data contained in the pixels are erased and the integration capacitors reset.

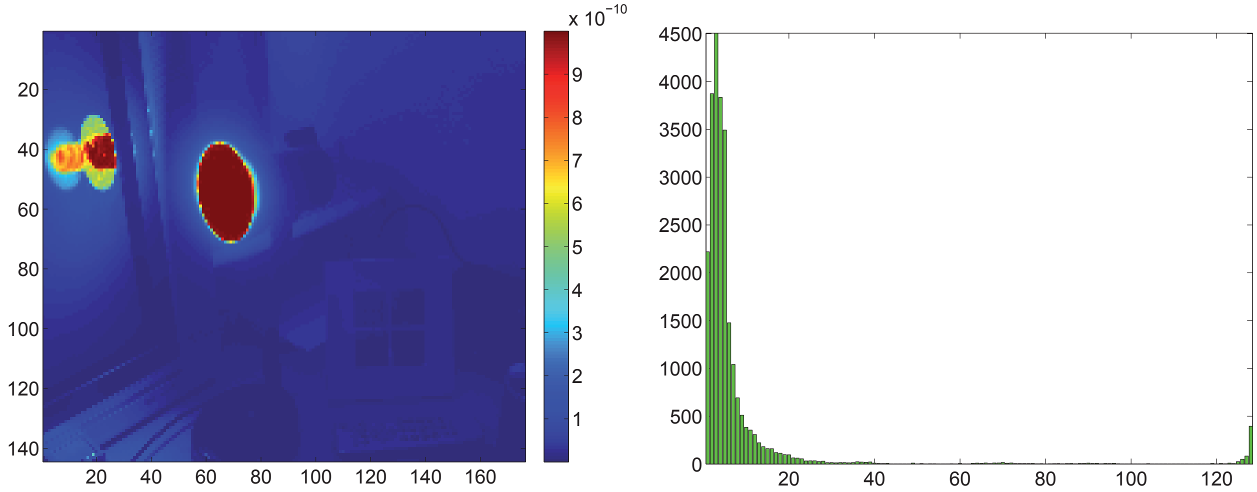
6. SIMULATION RESULTS

The operation of the proposed HDR technique is illustrated here using a highly bimodal HDR image as input. The starting point is a set of QCIF images which have been captured with different exposures. These images are combined afterwards in a 24-bit HDR standard representation using Adobe Photoshop'®. Photocurrent has been extrapolated from 1nA to 1fA, which implies a 120dB DR. Fig. 3(a) shows the normalized original image and (b) shows its conventional histogram (128 bins).

One can easily appreciate the extreme differences between the compact fluorescent lamp pixels, in red, and the poorly illuminated background, in blue, where almost nothing can be distinguished[¶]. As already mentioned, the first step is the configuration of the bins; for this example we have employed the configuration in Table 1. Since bins follow each other, the maximum exposure for this configuration is $31.1ms$. Afterwards, we simulate the temporal evolution of the different pixels in order to obtain V_{pixel} curves and the distribution of pixels among bins. Fig. 4(a) shows the resulting HI and Fig. 4(b) shows its histogram.

[§] Such that higher codes indicate higher illuminations, as in the usual codification of images.

[¶] The images are represented in matlab's colorjet colormap instead of gray for the sake of an improved visual perception of the details.

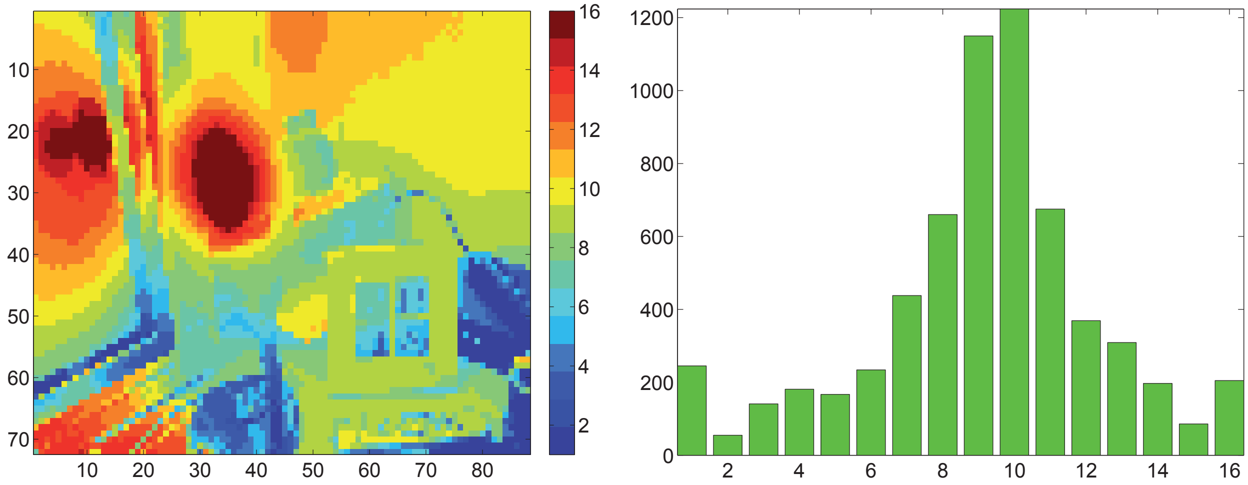


(a) *Scaled Original Image* (b) *Histogram*

Figure 3. Original Data.

The number of pixels in HI is 6336 (72·88). The levels per bin are assigned regarding the relevance of each bin into HI. Besides, since we are using 7-bit coding for TMI, the numbers of levels per bin are simply calculated by dividing the number of pixels in the bin by $\frac{6336}{128} = 49.5$. Now, the result must be floor-rounded in order to have an integer number of levels per bin, on the one hand, and to avoid distributing more than the available 128 codes, on the other hand. In this example this calculation leads to the distribution of 119 output codes among 16 bins. The unassigned levels are distributed among the bins according to the higher remainders in the division. These calculations are shown in Table 2, where bold letters indicate the assignation of the distributed levels.

The TMI curve is created by distributing these levels per bin uniformly within each bin. Hence, the tone-mapping curve is linear within each bin and its slope depends on both the levels assigned to the bin and the duration of that bin. Fig. 5(a) shows the resulting tone-mapping curve, where the vertical dotted lines are the limits between bins. Since the evaluation of whether the pixel voltage has crossed the reference signal is carried out 128 times during each bin, the total number of evaluations is $16 \cdot 128 = 2048$. Thus, the x-axis in Fig. 5(a) represents the value of the TMI curve in these evaluation times. The shape of the tone-mapping curve when the x-axis is the time has already been shown in Fig. 2. This curve is created decreasing codes when indicated by the LUT shown in Fig. 5(b), where black pixels denote no decrement and



(a) *Histogram Image* (b) *Cumulative Histogram*

Figure 4. Histogram Capture and Accumulation.

Table 2. Levels per bin calculus.

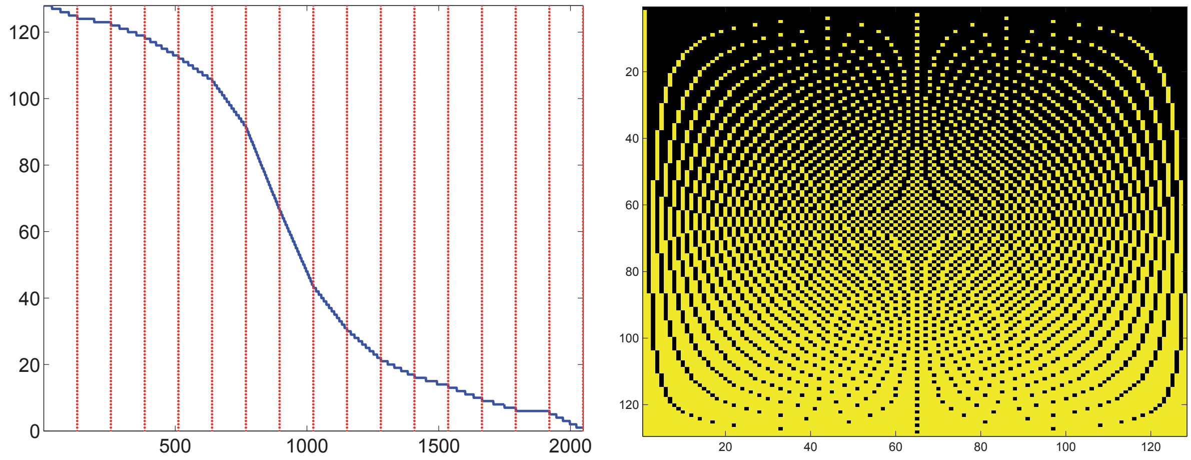
Bin index	Pixels in bin	$\frac{Pixels\ in\ bin}{49.5}$	Floor rounded levels	Remainder	Levels per bin
1	245	4.95	**4**	**47.0**	**5**
2	55	1.11	1	5.5	1
3	141	2.8485	**2**	**42.0**	**3**
4	181	3.6566	**3**	**32.5**	**4**
5	167	3.3737	3	18.5	3
6	234	4.7273	**4**	**36.0**	**5**
7	438	8.8485	**8**	**42.0**	**9**
8	660	13.3333	13	16.5	13
9	1150	23.2323	23	11.5	23
10	1224	24.7273	**24**	**36.0**	**25**
11	675	13.6364	**13**	**31.5**	**14**
12	369	7.4545	7	22.5	7
13	309	6.2424	6	12.0	6
14	197	3.9798	**3**	**48.5**	**4**
15	86	1.7374	**1**	**36.5**	**2**
16	205	4.1414	4	7.0	4

yellow pixels denote a one code decrement in the TMI curve. The y-axis indicates levels per bin(0-128) and x-axis depicts the evaluation index in this bin(from 1 to 128).

The final step is to simulate the acquisition of TMI using this TMI curve. The resulting TMI is shown in Fig. 6(a) and its histogram is displayed in Fig. 6(b). The effect of the algorithm is clearly noticeable here. Regarding the distribution of the codes, this algorithm creates an image with a fairly equalized histogram. Regarding the image details, despite photocurrents spread over 6 orders of magnitude, the image shows details in the poorly illuminated objects of the background, the spiral of the compact fluorescent lamp, and even the low-contrast numbers in the LCD clock.

7. CONCLUSIONS

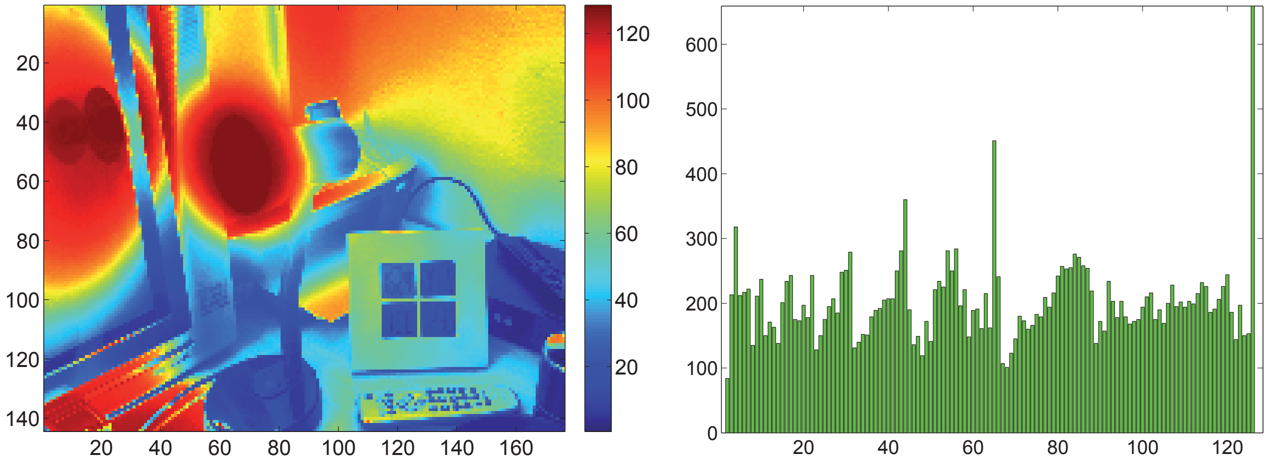
We have presented a Dynamic Range improving algorithm based on a tone mapping technique. The algorithm employs conventional integration pixels, comparators, and reduced local memory, thus being very suitable for Focal Plane Processor implementation. This technique reduces the amount of data in the HDR image to just 7-bit per pixel while preserving the relevant visual information. The algorithm adapts to the light conditions in only one frame, therefore it is very useful for



(a) Tone Mapping Curve

(b) Look Up Table

Figure 5. Tone Mapping Curve generation.



(a) *Tone Mapped Image* (b) *Tone Mapped Image Histogram*

Figure 6. Image produced by the algorithm.

HDR real-time smart cameras under moderate speed changing demands on the scene. A chip is currently being developed with the reported tone mapping system using Austrian Microsystems 0.35 microns technology.

ACKNOWLEDGMENTS

This work has been partially funded by TEC2009-11812 (WiVisNet), CENIT ADAPTA, V-MOTE 2006-TIC-2352, PIMA and FEDER 2007-2013.

REFERENCES

[1] Spivak, A., Belenky, A., Fish, A., and Yadid-Pecht, O., "Wide-dynamic-range cmos image sensors - comparative performance analysis," *Electron Devices, IEEE Transactions on* **56**, 2446 –2461 (nov. 2009).

[2] Sasaki, M., Mase, M., Kawahito, S., and Tadokoro, Y., "A wide-dynamic-range cmos image sensor based on multiple short exposure-time readout with multiple-resolution column-parallel adc," *Sensors Journal, IEEE* **7**(1), 151–158 (2007).

[3] Decker, S., McGrath, R., Brehmer, K., and Sodini, C., "A 256x256 cmos imaging array with wide dynamic range pixels and column-parallel digital output," in [*Solid-State Circuits Conference, 1998. Digest of Technical Papers. 45th ISSCC 1998 IEEE International*], 176–177, 433 (1998).

[4] Cheng, H.-Y., Choubey, B., and Collins, S., "An integrating wide dynamic-range image sensor with a logarithmic response," *Electron Devices, IEEE Transactions on* **56**, 2423 –2428 (nov. 2009).

[5] Erik Reinhard, Greg Ward, S. P. P. D., [*High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting*], Elsevier / Morgan Kaufmann (2006).

[6] Moini, A., [*Vision Chips*], Kluwer Academic Publishers, Norwell, MA, USA (1999).