High Dynamic Range Real-time Vision System for Robotic Applications

Pierre-Jean Lapray[1], Barthélémy Heyrman[1], Matthieu Rossé[1] and Dominique Ginhac[1,2]

Abstract— **Robotics applications often requires vision systems capable of capturing a large amount of information related to a scene. With many camera sensors, the perception of information is limited in areas with strong contrasts. The High Dynamic Range (HDR) vision system can deal with these limitations. This paper describes the HDR-ARtiSt hardware platform (High Dynamic Range Advanced Real-time imaging System), a FPGA-based architecture that can produce a real- time high dynamic range video from successive image acquisition. The proposed approach uses three captures with different exposure times, in a pure-hardware computationally efficient manner suitable for real-time processing. This method has the advantage to make outputs more visible than a simple single exposure. A real-time hardware implementation of High Dynamic Range video that shows more details in dark areas and bright areas of a scene is an important line of research. Our approach consists of three steps. First we capture three images from the sensor with alternating the three exposure times, and we store them into memory. Then we manage reading and writing operations in memory to have three video streams in parallel, corresponding to the three exposure times. Finally, under a highly parallel context, we blend the three video streams together with a modified version of the High Dynamic range technique. Our system can achieve an output at 60 fps with a full sensor resolution of $1,280 \times 1,024$ pixels. We demonstrate the efficiency of our technique through a series of experiments and comparisons.**

I. INTRODUCTION

The cameras have a limited dynamic range. In many videos, we have saturated zones in the dark and illuminated areas of images. These limitations are due to large variations of scene radiance, with over- and under-exposed areas in the single captured image. The sequential capture of several images with different exposure times can deal with the lack of information in extreme lightning conditions.

According to the book of Krawczyk et al. [1], there are two techniques to capture the entire dynamic of a scene. The first technique is to use an HDR sensor which, by design, is able to capture a wide dynamic range in a single capture. Several sensor types already exist like logarithmic sensors or sensors with local exposure adaptation[2]. Most of these specific sensors are also able to process raw images in order to extend the dynamic range[3], [4], [5], [6], [7], [8]. Unfortunately, such HDR sensors are often application tailor-made, expensive, not reusable in any context, and consequently not suitable for robotic applications. The other technique is to use a standard Low Dynamic Range (LDR) sensor and to perform HDR imaging from the LDR data. This technique derives several ways to proceed:

- Successive multiple capture on the same sensor with variable exposure times,
- Simple capture on a sensor with spatially varying masks in front of the sensor,
- Simultaneous simple capture on multiple sensors using a beam splitter.

According to A. E. Gamal [9], the multiple capture technique is the the most efficient method. Creating a multiple capture-based HDRi (High Dynamic Range imaging) system requires three steps:

- Recovery of the response curve of the system,
- Convert pixel into radiance values,
- Tone mapping.

This technique is designed to calculate the light intensities of real scenes, where each pixel is stored on a very large dynamic range (up to 32-bits wide and more). It is therefore necessary to have a large range of memory to store images, and then used them to reconstruct the final HDR image. Complex floating-point operators with a large data bus are also necessary to compute each value of radiance. The three most popular algorithms for HDR reconstruction are those of Debevec et al. [10], Mitsunaga et al. [11] and Robertson et al. [12]. A technical paper [13] compares the first two algorithms implemented in C/C++ in real time on PC. The results of computation time for an image are substantially the same.

HDR creating is followed by tone mapping operation (TMO). It is used to render the HDR data to match dynamic of conventional hardware display (mapped to [0,255]). There are two types of TMO: spatially uniform or global TMO and spatially non-uniform or local TMO. In our case, several algorithms can be implemented in real time due to fast computation capabilities, whether global or local. Here is a list of the most common algorithms, ordered from the fastest to the slowest software implementations:

- Durand et al. [14] (*Fast Bilateral Filtering for the Display of High-Dynamic-Range Images*)
- Duan et al. [15] (*Tone-mapping high dynamic range images by histogram novel Adjustment*)
- Fattal et al. [16] (*Gradient Domain High Dynamic Range Compression*)
- Reinhard et al. [17] (*Photographic Tone Reproduction for Digital Images*)
- Tumblin et al. [18] (*Time-dependent visual adaptation for fast realistic image display*)
- Drago et al. [19] (*Adaptive Logarithmic Mapping For Displaying High Contrast Scenes*)

Only the TMO Reinhard et al. [17] method has already been

[1]P.J. Lapray, B. Heyrman, M. Rossé and D. Ginhac are with Le2i UMR 6306 - University of Burgundy, Dijon, France

[2]D. Ginhac is the corresponding author `dginhac at u-bourgogne.fr`

the subject of hardware real-time work. Unfortunately, the algorithm needs whole images at input to work and requires a large amount of memory. Based on a exhaustive study of these different methods, we decided to use the global tone mapper introduced by Duan et al. [20].

In the remainder of is paper, we propose a complete parallel architecture and an associated hardware implementation on a FPGA-based platform capable of producing real-time HDR content. This contribution to real-time video processing can be easily applied to several environments and more specifically to robotic applications. In Section II, we only focus on HDR methods that are suitable for a real-time implementation, and more specifically on on the original algorithm developed by Debevec et al. [21] for HDR creating and the global tone mapping operator proposed by Duan et al. [20]. Based on this deep study, we propose a pure-hardware implementation on our FPGA-based platform in Section III. Some experiments and results follow this discussion in Section IV. Concluding remarks are then presented.

II. Algorithm for HDR vision system

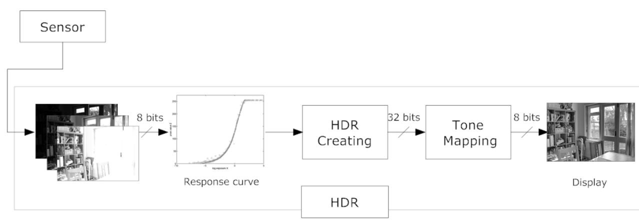


Fig. 1. High Dynamic Range pipeline.

A. Step 1: HDR creating

With standard imaging systems, we can generally produce 8-bits, 10-bits or 12-bits images. Unfortunately, this does not cover the full dynamic range of irradiance values in real scene. The most common method to generate HDR content is to capture multiple images with different exposure times. If the camera has a linear response, we can easily recover the irradiance E_i with each radiance Z_{ip} and exposure times Δt_p stored in each exposure p. However, cameras do not provide a linear response. Consequently, we have to evaluate as precisely as possible the response curve of the vision system to combine properly the different exposures. So, creating an HDR image from multiple LDR images, i.e. producing a radiance map requires two major stages:

- Recovering the sensor response function;
- Reconstructing the HDR radiance map from multiple exposures.

From literature, three most popular algorithms for recovering the response curve (g) can be extracted: Debevec et al. [21], Mitsunaga et al. [22], and Robertson et al. [12]. A technical paper [23] compares the first two algorithms in a real-time software implementation. The results of time calculation for an image are significantly identical. Based on a detailed description of these methodologies[24], we decided to use the original algorithm developed by Debevec et al. [21].

From the system response curve g, the radiance value of each pixel can be evaluated by a weighted combination of the pixels from each LDR frame. According to Debevec et al. [21], the film reciprocity equation is:

$$Z_{ij} = f(E_i \Delta t_p) \tag{1}$$

Where E_i is the irradiances, Z_{ip} is the pixel value of pixel location number i in image p and Δt_{ip} is the exposure duration. The function g is defined as $g = \ln f^-1$. The response curve g can be determined by resolving the following quadratic function:

$$\mathcal{O} = \sum_{i=1}^{N} \sum_{p=1}^{P} [g(Z_{ip}) - \ln E_i - \ln \Delta t_p]^2 + \lambda \sum_{z=Z_{min}+1}^{Z_{max}-1} g''(z)^2 \tag{2}$$

Where $g(z)$ is the log exposure corresponding to pixel value z and λ is a weighting scalar depending on the amount of noise expected on g. This leads to 1024 values of $g(z)$, minimizing the equation 2. The response curve can be evaluated before the capture process, from a sequence of several images. The curve can be used to determinate radiance values in any image acquired during the video streams. For Schubert et al. [25], the HDR reconstruction requires only two images. The following equation can be used:

$$\ln E_i = \frac{\sum_{p=1}^{P} \omega(Z_{ip})(g(Z_{ip}) - \ln \Delta t_{ip})}{\sum_{p=1}^{P} \omega(Z_{ip})} \tag{3}$$

Where $\omega(z)$ is the weighting function. It is a simple hat equation corresponding to the weighting function $\omega(z)$:

$$\omega(z) = \begin{cases} z - Z_{min} \text{ for } z \leq \frac{1}{2}(Z_{min} + Z_{max}) \\ Z_{max} - z \text{ for } z > \frac{1}{2}(Z_{min} + Z_{max}) \end{cases} \tag{4}$$

B. Step 2: Tone mapping

The HDR pixels are represented by a high bit-depth conflicting with the standard display devices, requiring a high to low bit-depth tone mapping. Cadik et al. [26] show that the global part of a tone mapping operator is most essential to obtain goods results. A psychophysical experiment by Yoshida et al. [27], based on a direct comparison between the appearances of real-world HDR images shows that global methods like Drago et al.[28] or Reinhard et al.[17] are perceived as the most natural ones. Moreover, a global tone mapper is the easiest way to reach real-time constraints, because local operators require complex computations, and also may generate halo artifacts. The choice of a candidate tone mapping operator has been done after comparing some global algorithms applied to a radiance map constructed from two or three images. Several global algorithms have been extensively tested in C++. According to our temporal and hardware constraints, the best compromise is the global tone mapper introduced by Duan et al. [20]. Here is the equation for tone mapping:

$$D(E_i) = C * (D_{max} - D_{min}) + D_{min}$$
$$with\ C = \frac{\log(E_i + \tau) - \log(E_{i(min)} + \tau)}{\log(E_{i(max)} + \tau) - \log(E_{i(min)} + \tau)} \tag{5}$$

where D is the displayable luminance and E_i a radiance value of the HDR frame. $E_{i(min)}$ and $E_{i(max)}$ are the minimum and maximum luminance of the scene and τ is the overall brightness control of the tone mapped image.

III. Real-time implementation of the HDR vision system

A. The HDR-ARtiSt platform

We built an FPGA-based smart camera built around a Xilinx ML507 development board (including a Virtex-5 FPGA) and a specific daughter board hosting a sensor provided by e2v (1.3 million pixels CMOS image sensor [29]) as seen on Fig. 2. With a dynamic range around 60dB, this sensor can be considered as a low dynamic range sensor. The DVI output of the FPGA board is connected to an external LCD monitor to display the HDR images.

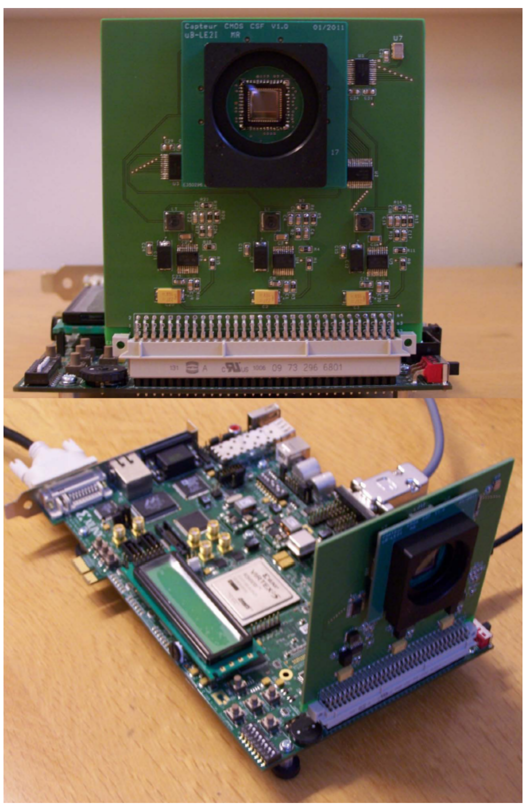


Fig. 2. The hardware architecture: an FPGA (ml507 Xilinx card) and a daugther card with the e2v sensor

B. Memory architecture

The sensor is able to send full-resolution images at 60 frames/s. Images successively captured with long, middle and short exposures are first stored in DDR2 memory and then read in parallel to create the HDR image with the methods described in subsections II-A and II-B. It's important to notice that the design is a pure-hardware system which is processor-free and must be able to absorb a continuous pixel flow of about 80 Mega pixel per second from the sensor (called "DDR2 In" in Fig. 3) while reading two other pixel flows corresponding to the two stored images (respectively called "DDR2 Out1" and "DDR2 Out2" in Fig. 3).

Considering the three first consecutive images I_1, I_2, and I_3 captured with three different exposure times (long, middle and short exposures) as illustrated on Fig. 3. During the capture of the first frame (I_1), the image is stored line by line (WL_jI_1, with $1 <= j <= 1024$). For the second frame I_2, the image is also stored line by line (WL_jI_2) while the first frame is read from the memory (RL_jI_1). Then, during the capture of the last frame (I_3), lines of the two previous frames stored are synchronously read from the memory during inter-frame (RL_jI_1, RL_jI_2) and buffered into BRAM while each new captured line (WL_jI_3) is stored in memory. The HDR process needs a continuous stream of pixels of three images and then can only be performed while receiving the third frame. Then, the process can continue in the same way throughout the capture of the fourth frame (I_4) and the readout of the second and third frame (I_2 and I_3). Finally, our memory management system is able to deliver two parallel pixel streams that have been acquired and stored into the memory and a third pixel stream directly from the sensor. With this technique, each HDR pixel only requires three memory accesses (one write and two read operations during one line interval), saving many memory access operations. The main advantages of such a technique are to store only two images in memory, avoid the waiting for the three images to compute an HDR image and deliver an HDR image stream at the framerate of the sensor.

C. Algorithm details

According to the Section II), our HDR vision system is based on the Debevec et al. [21] algorithm for the HDR creating step and the Duan et al. [20] method for the tone mapping step. However, in order to cope with the hardware and the temporal constraints, some adjustments and simplifications have to be proposed.

For the Debevec et al. [21] algorithm, the response curve g is evaluated during a initialization step with a dedicated software and then stored in LUTs on the hardware platform, as seen in Fig. 4. This step needs to be performed only once for each new camera that is plugged into the FPGA board. Moreover, to reduce the computation complexity and to optimize area utilization, some operations, like neperian logarithms, are also pre-calculated and registered in LUTs. Finally, the hardware cost of the implementation of HDR creating obviously depends on the number of bracketed images used to create the HDR image but also the computations performed on these images. As an illustration, the HDR creating pipeline depicted in Fig. 4 requires three 32-bits multipliers, one 32-bits divider, and four transitions from 8-bits to IEEE754 32-bits wide, named *Fixed-to-Float*.

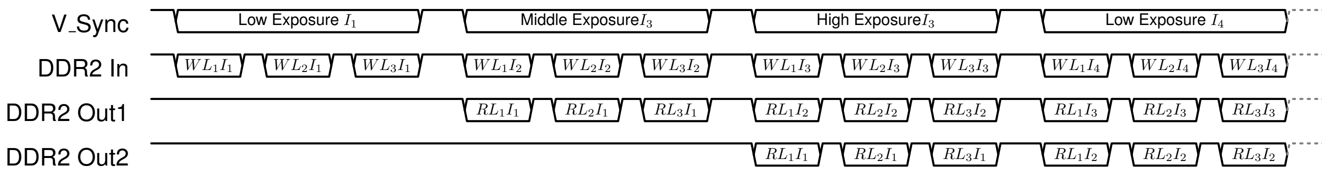


Fig. 3. Performing three streaming videos (high, middle and low exposure videos) in parallel. SDRAM is used to line writing and reading operations.

For the tone mapping operator, the evaluation of the equation (see Section II-B) requires to have a full HDR stored frame. However, as our hardware pipeline performs HDR imaging on the pixel stream, the terms $E_{i(min)}$ and $E_{i(max)}$ are computed from the previous HDR frame and not from the current frame (respectively by the ”CMP min” and the ”CMP max” operators on the Fig. 5). This choice can be easily justifiable because we can make the assumption that two consecutive frames captured by the sensor have approximatively the same properties in terms of brightness.

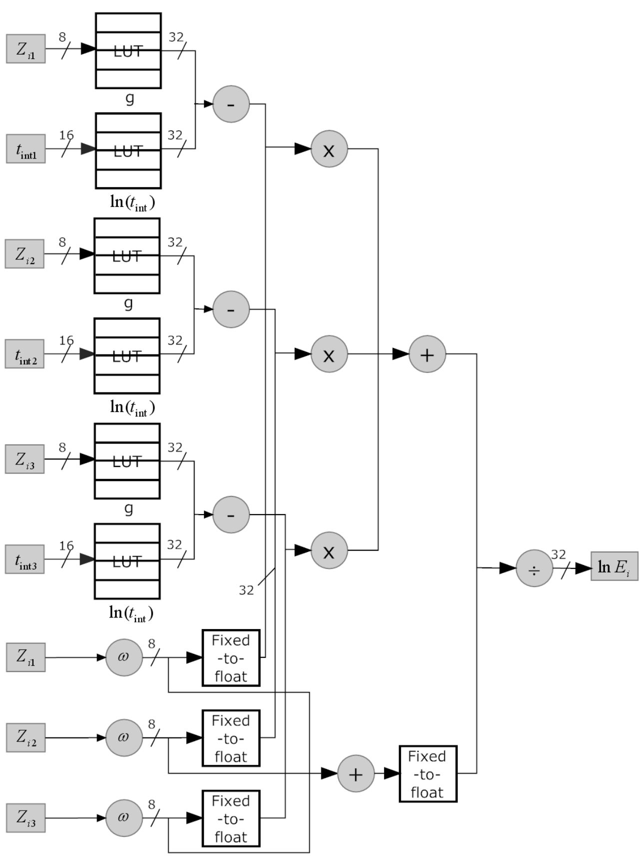


Fig. 4. HDR creating pipeline for HDR video, using LUTs tree.

IV. Experimental Results

The proposed architecture has been checked using Modelsim and then implemented on the HDR-ARtiSt platform described in Section III-A delivering a real-time 1280×1024 resolution HDR video at 60 frames/s. Table I summarizes the synthesis report and reveals a low use of logic cells.

An example of a resulting HDR image is shown on Fig. 6 obtained from a set of low, middle and high exposure frames

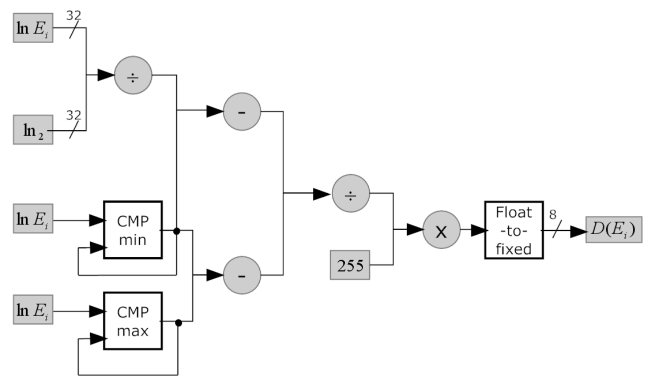


Fig. 5. Tone mapping computation.

TABLE I

HDR vision system with three exposures: summary of hardware synthesis report on Virtex-5.

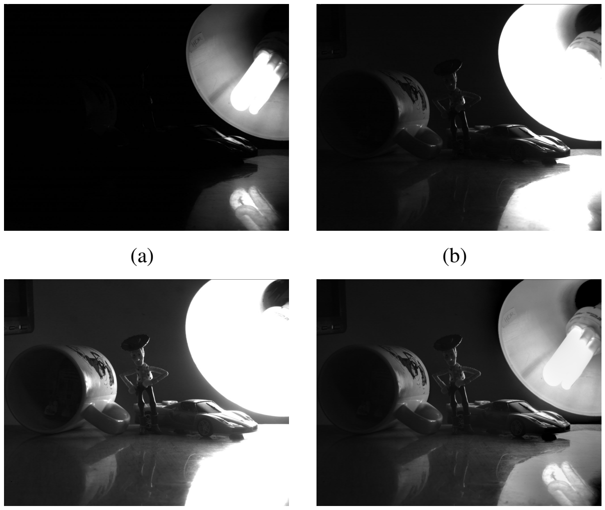
	Used	Available	Utilization
Logic utilization			
Number of Slice flip flops	15,391	44,800	34%
Number of Slice LUTs	13,441	44,800	30%
Complexity distribution			
Number of occupied slices	6,222	11,200	55%
Memory management	505		4%
HDR creating	2,466		22%
Tone mapping	1,024		9%
Post processing	1,041		9%
Number of FIFO/RAMBs	18	148	12%
Memory management	8		5%
HDR creating	4		2%
Tone mapping	0		0%
Post processing	7		4%
Number of DCM-ADVs	4	12	33%

(respectively Fig. 6-a, Fig. 6-b and Fig. 6-c). Details in dark areas and bright are represented in our tone mapped image without artifacts.

V. Conclusion

In this paper, we present a full-hardware architecture dedicated to HDR vision that can produce high dynamic content in real-time, and satisfy all constraints and image quality requirements. The model implemented here can be easily adapted to any FPGA-based smart camera providing images of different dynamic ranges or resolutions.

Our system can achieve an output at 60 fps with a 1280×1024-pixel resolution. It is able to reconstruct high-quality



(a) (b) (c) (d)

Fig. 6. A multiple exposure sequence (exposure times: (a) $0.15ms$, (b) $0.78ms$ and (c) $1.9ms$) of three bracketed images, the HDR output of the hardware platform (d).

HDR contents from three captures using the conventional technique of HDR creating and a global tone mapping.

We plan to develop and benchmark others variety of state-of-art algorithms both for computing radiance maps, calibrating HDR images, and local/global tone mapping.

REFERENCES

[1] R. M. Karol Myszkowski and G. Krawczyk, *High Dynamic Range Video*, B. A. B. S. Editor, Ed. Morgan et Claypool Publishers, 2008.

[2] S. Kavusi and A. El Gamal, "A quantitative study of high dynamic range image sensor architectures," in *Proceedings of the SPIE Electronic Imaging '04 Conference*, vol. 5301, Jan 2004, pp. 264–275.

[3] P. Acosta-Serafini, M. Ichiro, and C. Sodini, "A 1/3" VGA linear wide dynamic range CMOS image sensor implementing a predictive multiple sampling algorithm with overlapping integration intervals," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 9, pp. 1487–1496, Sept 2004.

[4] G. Cembrano, A. Rodriguez-Vazquez, R. Galan, F. Jimenez-Garrido, S. Espejo, and R. Dominguez-Castro, "A 1000 FPS at 128 × 128 vision processor with 8-bit digitized I/O," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 7, pp. 1044–1055, Jul 2004.

[5] M. Sakakibara, S. Kawahito, D. Handoko, N. Nakamura, M. Higashi, K. Mabuchi, and H. Sumi, "A high-sensitivity CMOS image sensor with gain-adaptative column amplifiers," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 5, pp. 1147–1156, May 2005.

[6] L. Lindgren, J. Melander, R. Johansson, and B. Mller, "A multiresolution 100-GOPS 4-Gpixels/s programmable smart vision sensor for multisense imaging," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 6, pp. 1350–1359, Jun 2005.

[7] S. Morfu, P. Marqui, B. Nofil, and D. Ginhac, "Chapter 3 nonlinear systems for image processing," in *Advances in IMAGING AND ELECTRON PHYSICS*, ser. Advances in Imaging and Electron Physics, P. W. Hawkes, Ed. Elsevier, 2008, vol. 152, pp. 79 – 151. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1076567008006034

[8] J. Dubois, D. Ginhac, M. Paindavoine, and B. Heyrman, "A 10 000 fps CMOS sensor with massively parallel image processing," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 3, pp. 706–717, Mar 2008.

[9] A. E. Gamal, "High dynamic range image sensors," in *ISSCC*, 2002.

[10] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *SIGGRAPH*, 1997, pp. 369–378.

[11] T. Mitsunaga and S. Nayar, "Radiometric Self Calibration," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, Jun 1999, pp. 374–380.

[12] M. A. Robertson, S. Borman, and R. L. Stevenson, "Estimation-theoretic approach to dynamic range enhancement using multiple exposures," *Journal of Electronic Imaging*, vol. 12(2), pp. 219–228, 2003.

[13] W. S. G. Yourganov, "Acquiring high dynamic range video at video rates," Acquiring High Dynamic Range Video at Video Rates, Tech. Rep., 2001.

[14] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," *ACM Trans. Graph.*, vol. 21, pp. 257–266, July 2002.

[15] C. G. Jiang Duan, MarcoBressan, "Tone-mapping high dynamic range images by novel histogram adjustment," *Pattern Recognition*, vol. 43, pp. 1847–1862, 2010.

[16] M. W. Raanan Fattal, Dani Lischinski, "Gradient domain high dynamic range compression," School of Computer Science and Engineering The Hebrew University of Jerusalem, Tech. Rep., 2002.

[17] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, "Photographic tone reproduction for digital images," *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 267–276, 2002.

[18] S. N. Pattanaik, J. Tumblin, H. Yee, and D. P. Greenberg, "Time-dependent visual adaptation for fast realistic image display," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, ser. SIGGRAPH '00. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 2000, pp. 47–54. [Online]. Available: http://dx.doi.org/10.1145/344779.344810

[19] T. A. F. Drago, K. Myszkowski and N. Chiba, "Adaptive logarithmic mapping for displaying high contrast scenes," *EUROGRAPHICS 2003 / P. Brunet and D. Fellner*, vol. Volume 22 (2003), Number 3, no. 3, pp. xx – xx, 2003.

[20] J. Duan, M. Bressan, C. Dance, and G. Qiu, "Tone-mapping high dynamic range images by novel histogram adjustment," *Pattern Recognition*, vol. 43, pp. 1847–1862, May 2010.

[21] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *Proceedings of the 24th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 1997, pp. 369–378.

[22] T. Mitsunaga and S. Nayar, "Radiometric Self Calibration," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, Jun 1999, pp. 374–380.

[23] W. S. G. Yourganov, "Acquiring high dynamic range video at video rates," Acquiring High Dynamic Range Video at Video Rates, Tech. Rep., 2001.

[24] M. Granados, B. Ajdin, M. Wand, C. Theobalt, H. Seidel, and H. Lensch, "Optimal hdr reconstruction with linear digital cameras," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 215–222.

[25] F. Schubert, K. Schertler, and K. Mikolajczyk, "A hands-on approach to high-dynamic-range and superresolution fusion," in *IEEE Workshop on Applications of Computer Vision (WACV)*, 2009, pp. 1–8.

[26] M. Čadík, M. Wimmer, L. Neumann, and A. Artusi, "Evaluation of HDR tone mapping methods using essential perceptual attributes," *Computers & Graphics*, vol. 32, pp. 330–349, 2008.

[27] A. Yoshida, V. Blanz, K. Myszkowski, and H. peter Seidel, "Perceptual evaluation of tone mapping operators with real-world scenes," in *Human Vision & Electronic Imaging X, SPIE*. SPIE, 2005, pp. 192–203.

[28] T. A. F. Drago, K. Myszkowski and N. Chiba, "Adaptive logarithmic mapping for displaying high contrast scenes," *EUROGRAPHICS 2003 / P. Brunet and D. Fellner*, vol. Volume 22 (2003), Number 3, no. 3, pp. xx – xx, 2003.

[29] e2v technologies, "Ev76c560 BW and colour CMOS sensor," 2009. [Online]. Available: http://www.e2v.com/products-and-services/high-performance-imaging-solutions/imaging-solutions-cmos-ccd-emccd/