Analysis and Development of Tone Mapping Algorithms on High Dynamic Range Color images: Tone Mapping

Tooba Shams, Santiago Herrero

***Abstract*—Classic digital images miss certain details of a same scene due to the light exposure they are subjected to or due to device limits. High Dynamic Range images are a very appealing aspect in photography that allows us to perceive all those details of a scene in the same picture, by combining images of the same scene but acquired with a different light exposure. This means that we can obtain a "perfect" image in which all relevant details are visible at the same time.**

The whole process consists of obtaining High Dynamic Range (HDR) color images and developing algorithms, called tone mapping algorithms, for visualizing such images on current existing devices (Low Dynamic Range -LDR- ones).

The work presented in this paper consists of the study and implementation of algorithms for visualizing already generated HDR images in LDR devices. For this purpose, we have used very different HDR images, including the ones generated by ourselves in [6]. The different techniques performed goes from very simple and naif implementations, to much more sophisticated ones, including a study of their behaviour in different color spaces.

I. Introduction

From personal mobile phone photograph to emerging image processing fields, digital photography plays a fundamental role in current days. Digital images, in general, are the converted values of the analog information of a real scene. This conversion causes information loss and a lot of details are not visible in the resulting images. These images have low dynamic range of color.

A High Dynamic Range image shows the radiance values of the image that can not be captured from an ordinary imaging device or photographic techniques. An HDR image aims to provide a similar range of radiance that is visualized by the human eye. However, imaging devices, have a restriction when it comes to capturing these values of radiance, and we also find problems when trying these images to be displayed by different display or printing devices. For this reason the values of an HDR image are mapped to make them compatible with such devices. This process is called Tone mapping.

Therefore, the whole process can be broken down into two main phases:

- Generation of High Dynamic Range Images.
- Developing and analyzing tone mapping algorithms on HDR color images.

Digital devices can only capture a certain range of radiance values. In modern cameras, the range can be chosen depending on the values required. However, capturing the actual range of the whole scene still remains a problem. Further scenes with high amount of sunlight or very dark scenes can be a problem when it comes to capture the actual dynamic range of the scene due to the saturation problems of the camera sensors. These problems can be solved by taking several photos of the same scene and obtaining a High Dynamic Range Image. In TRDP-1, we describe the algorithm we have implemented, based on [2], to obtain this High Dynamic Range image, by finding the underlying response of the imaging system. This was simply done with several images of the same scene and the information of the exposure time for each image. We also proposed methods to obtain optimal values of different parameters involved in the process finally, providing a detailed discussion over the results and parameters involved. The details can be found in [6].

This part of the TRDP consists of mapping the previously obtained HDR image. As explained in [3], the dynamic range illumination in the real world is much higher than the dynamic range that can be displayed by existing devices (around 100 times higher), which becomes even larger in scenes with high contrast (for example, a dark indoor room receiving outdoor light through a window). For this reason, we need to create a match between the real observed scene and the image displayed. This match is performed by the Tone Mapping algorithm.

It is not enough to compress the real luminance values to the scale of the displaying device, as this leads to big contrast and loss of detail. For a better understanding on how to develop these Tone Mapping algorithms, we should take into account how the Human Visual System (HVS) works, as it functions over a huge range of illumination, especially regarding the photo-receptors adaptation ability. This is based on the neural responses by the rods and cones when absorbing light.

The rest of the paper is organized as follows: Section II shows a brief summary of the existing solutions for this problem and the related documentation. Section III makes a brief summary of what is required during this study. Section IV shows the work process during the semester. Section V portrays a quick overview of the images used during the course of the study and how the code was developed. Section VI presents the different methods performed to obtain the goal. Section VII presents and discusses experimental results leading to a set of conclusions in section VIII.

II. State Of The Art

The first part of this project consisted of generating an HDR image. This part was developed in UAM. The generation of HDR images can be done using a set of images of the same scene with different exposures. The generated HDR image can be considered as the analog image, containing the minor details of the scene as mentioned in [1]. The idea proposed by the authors is to use the integer images to generate a floating point representation, which can be considered as the analog image, having a much higher dynamic range. This paper served as a basis for understanding HDR images.

In the generation of an HDR image the set of images can be used to determine the underlying nonlinear response of the imaging system as mentioned in [2]. The authors used the images to find this response curve, hence, achieving the values of the final HDR image. The approach in [2] is described over grayscale imaging device and some hints are given for generalizing the algorithm for color images. The physical properties of the imaging system can be exploited to find the original radiance of the pixels as mentioned in [2]. The authors present a robust and accurate method for the recovery of a High Dynamic Range image. However, the performance of different parameters still needs to be explored in order to get the best results. The documentation of the generation of an HDR image can be seen in [6].

This part of the TRDP includes the tone mapping of the generated HDR images. The display of the contrast between very illuminated and very dark sections of a same image has always been a problem to be addressed in photography. Tone Mapping algorithms are the mathematical answer to these problems. Based on how the HVS tackles the contrast issues, several algorithms have been developed with successful results.

The most basic algorithm, presented in [3], is the *Naka-Rushton* equation, which models an S-shaped function based on how rods and cones of a human eye work. The *Michaelis-Menten* equation is an extension of this first one. The authors in [3] also propose two different averaging of the image, in order to obtain more realistic results.

In [4] and [5], the authors present a more complex solution, which is performed individually for every pixel of the HDR image, which makes it more computationally expensive than the previously presented ones.

III. Requirements

This part of the TRDP requires the following:

- Learning the human visual system properties associated with tone mapping
- Implementation of a naive tone mapping operator
- Problems of this operator and proposal of solutions
- Effect of tone mapping operator on color space.
- Further analysis

IV. Gantt Chart

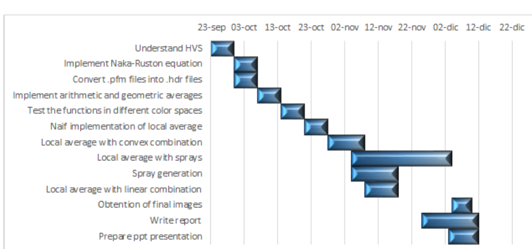
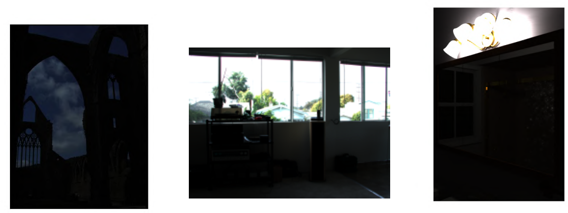


Fig. 1: Gantt chart

V. Software Architecture

For developing the code and testing the different results, we have used a small but sufficient dataset of thirteen images, to which we have added the self-generated HDR images in [6]. These images present different scenarios of high contrast, differently illuminated scenes. These scenes vary from indoor-outdoor light contrast (2b), to indoor differently illuminated images (2c) or outdoor images with highly contrasted sections (2a). Some of the images had to be converted into *.hdr* format, as they were provided in *.pfm* format, which cannot be read by MATLAB.

The full code has been developed in MATLAB, in different sections depending on the color space that the algorithms are applied to. For each color space, there is a subdivision depending on the approach taken.



(a) tinterna.hdr

(b) apartment.hdr

(c) rend04_o80A.hdr

Fig. 2: Sample HDR images without any tone mapping

VI. Implementation

Our implementation consists of coding all the algorithms from scratch with the help of some state of the art knowledge. We further extend this knowledge by implementing and trying further mathematical formulations. The first part of this TRDP [6] explains all the implementation done to generate good quality results, but the implementation process described in this section only focuses over the tone mapping algorithms created and tested during TRDP 2. An overall implementation can be seen from the block diagram shown in Fig. 3.

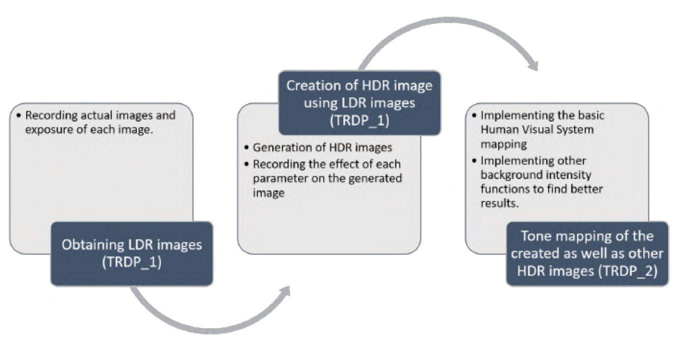


Fig. 3: TRDP Block Diagram

A. The Human Visual System HVS

As introduced in the beginning, a problem with low dynamic range images is that they can not contain all the dynamic range of the scene. An HDR image on the other hand has information proportional to the actual dynamics of the scene. However this HDR image without any tone mapping cannot be visualized on a display device due to the wide dynamic range. The tone of such an HDR image needs to be mapped to display it. This mapping can be imagined as if passed each pixel of an image through a function to obtain new values for it. The question one might ask is what is this function. This function will be termed as a tone mapping operator. To create a basic tone mapping operator, the working of the Human Visual System (HVS) can be considered due to its capability of processing huge range of illuminations. This however happens because the HVS adapts itself depending on the illumination conditions. For instance during daylight, HVS is less sensitive to small sources of lights and ignores them however at night one can easily see a distant light bulb. This amazing ability of the HVS to adapt to the range of illumination serves another ground to implement a tone mapping operator based on HVS. To this, it is essential to understand the working of the HVS, for which we have based our research in chapter 7 of [3].

The HVS mechanism can be broken down into two processes. The pupil action and the photo-receptor mechanism. The pupil action can be summed up as the change of diameter of the pupil depending on the illumination conditions. For example, the size of the pupil is bigger during the night, so that more light can enter it and vice versa. However, the photo-receptor mechanism on the other is more important and serves as the building block of our implementation.

The photo-receptors are the cells converting light into neural signals. These receptors are of two types called the rods and the cones. The rods are more sensitive to light and are more active during the night, this response is called the scotopic response. On the other hand the cones are less sensitive and work during the day or in brighter illumination conditions, creating the photopic response. The response of both the receptors overlap and is known as the mesopic response. The combined response curve of Rods and Cones can be visualized in Fig. 3. taken from chapter 7 of [3].

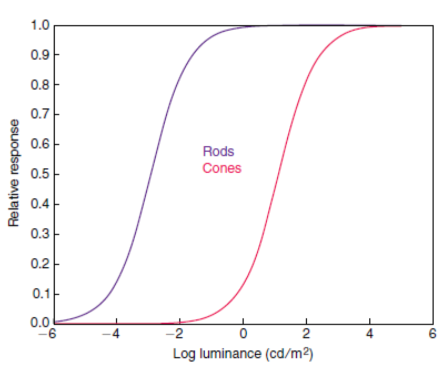


Fig. 4: The response of Rods and Cones cells to various intensities in arbitrary units.

This response curve can be fitted using the Naka-Rushton Equation stated in (1).

$$\frac{R}{R_{max}} = \frac{I^n}{I^n + \sigma^n} \tag{1}$$

where R is the photo-receptor response (0 < R < Rmax), R_{max} is the maximum response, I is light intensity, and σ is the semisaturation constant where the intensity causes the half-maximum response.

The shapes of the intensity-response curves are independent of the background intensity. However, with background, the position of the response function shifts horizontally along the intensity axis. This shows that the HVS always maintains its long linear property if given the time to adapt. This shift is modeled by increasing the value of the semi saturation constant as a function of the background intensity. Hence, the equation becomes:

$$\frac{R}{R_{max}} = \frac{I^n}{I^n + \sigma_b^n} \tag{2}$$

where

$$\sigma_b = (f I_b)^m \tag{3}$$

where I_b is the background intensity of the scene.

Throughout our implementation we have considered f=m=1. Equations (2) and (3) serve as the basic ground of our implementations.

B. Experimental implementation

The first task consisted of implementing (2) with (3) as our chosen constraints. Each pixel value I^n of an HDR image undergoes (2) to obtain a new pixel value R.

A very important part of the implementation is to calculate the background intensity I_b of each test image. The most basic idea is to take the mean of the entire image and use it as I_b. This mean can be chosen as either arithmetic or geometric mean. This kind of background intensity calculation uses the global properties of an image and hence corresponds to the global mean. To do the global average of the image, we created two functions. One of the functions calculates the arithmetic mean of the pixel values of the HDR image while the other function returns the geometric mean. The implementation is done using the following equations.

For arithmetic mean:

$$I_b = \frac{1}{N}\sum_{p=1}^{N} I_p \tag{4}$$

For geometric mean:

$$I_b = \prod_{p=1}^{N} (I_p + \epsilon)^{(\frac{1}{N})} \tag{5}$$

where p is the number of pixels in the HDR image and ϵ is a very small increment to avoid non zero pixel values. If this value is too big, the intensity input values are falsified and the resulting image would be much darker than it should be.

Another way to get the background intensities is to explore the HDR image locally. We used different techniques to manipulate the local properties of the HDR image in order to calculate the background intensity function. The methods used are listed below.

- **Kernel Based Approach:** In this approach, we calculate the mean of the pixels in an area using a kernel. The idea is to slide this kernel along the image to calculate the background intensities of different regions. To do this we used square kernels with a size equal to the greatest common divisor of the dimensions of the HDR image. The idea of finding the greatest common divisor is due to the observation of bad quality results obtained for small sized kernels. The obtained mean of a specific region is compared with the global arithmetic mean in order to choose between the arithmetic or geometric mean for that specific region. The comparison can be summed up by equations (6) and (7).

$$mean_k = \frac{1}{N}\sum_{p=1}^{N} I_p \tag{6}$$

$$I_{b,k} = \begin{cases} \prod_{p=1}^{K}(I_p+\epsilon)^{(\frac{1}{K})}, & \text{if } mean_k \leq I_b \\ \frac{1}{K}\sum_{p=1}^{K} I_p, & \text{otherwise} \end{cases} \tag{7}$$

  where I_b is the arithmetic mean of the entire HDR image calculated globally, $I_{b,k}$ is the final background intensity of a region k and $mean_k$ is the arithmetic mean of a region k, based on the kernel.
  The condition in equation (7) can be explained through the following fact in equation (8).

$$\mu_a \geq \mu_g \tag{8}$$

  Where μ_a and μ_g correspond to the arithmetic and geometric mean respectively. This mean is used in the denominator of equation (2). This means that the geometric mean being smaller will yield a brighter image as the division is by a smaller number. Contrary to this, an image obtained by using the arithmetic mean will be darker. Hence the idea in equation (7) is to push the dull parts of the image to become brighter while keeping the brighter parts unsaturated.
- **Convex combination of arithmetic and geometric mean:** In this method we used a combination of the arithmetic and geometric mean for each pixel, which is normalized respect to the greatest value of the image.

$$\sigma_p(i,j) = (1 - I(i,j))\mu_g + I(i,j)\mu_a \tag{9}$$

  The goal is to push the darker pixels towards lighter values without saturating the lighter pixels. According to equation (9), if the mean of a pixel is lighter, the arithmetic mean is given more weight over the geometric mean and vice versa.
- **Linear combination of arithmetic and geometric mean:** Equation (9) did not give very good results in the end due to the fact that it never allows to apply just the geometric averaging to very dark pixels. By expanding equation (9), where we will now have $I(i,j) = \epsilon$,

$$\begin{aligned} \sigma_p &= (1-\epsilon)\mu_g + \epsilon\mu_a \\ &= \mu_g - \epsilon\mu_g + \epsilon\mu_a \\ &= \mu_g + \epsilon(\mu_a - \mu_g) \end{aligned}$$

  where, being $\epsilon > 0$ and $\mu_a > \mu_g > 0$ always, then $\sigma_p > \mu_g$ always.
  We moved towards a linear combination of the arithmetic and geometric mean to implement the same goal as explained in the previous sections. This approach allows a slight smoothing of the image in order to avoid outliers. Equation (10) shows the implementation of this method

$$\sigma_p = \mu_g + \frac{(\mu_a - \mu_g)}{I_{max} - I_{min}}(I - I_{min})^{\gamma} \tag{10}$$

  where I_{max} is the maximum intensity value, I_{min} is the minimum intensity value, I is a pixel value and $\gamma > 1$ is a factor added to change the shape of the equation. This action is performed in each of the channels.
  The parameter γ can be used to change the equation from linear to convex or concave by simply changing its value. For a linear equation the value of γ is 1. In this case, when $I = I_{max}$, only the arithmetic average will be computed for that pixel, and when $I = I_{min}$, only the geometric average will be computed for that pixel.
  This method gave us better results which were fast and also removed the problems of the previous methods.
- **Convex combination of arithmetic and geometric mean combined with sprays:** In order to improve the obtained results with the convex implementation, we decided to apply the sprays technique. Following the spray generation described in [4], we created our own kernel based on sprays and applied them in the image. For generating the spray, a circle of radius 1, centered in ($i_x = 1, i_y = 1$) is created and filled with a variable amount of points. Examples of sprays with different amount of points can be seen in Fig. 5.
  These points conform the spray equation (11) based on the random values of ρ (in the interval [0,1]) and θ (in the interval [0,2π]).

$$\begin{cases} j_x = i_x + \rho cos(\theta) \\ j_y = i_y + \rho sin(\theta) \end{cases} \tag{11}$$

  where j_x and j_y take values in the interval [0,2] and are made proportional to the size of the kernel. According to the work developed in [4], instead of using ρ itself it is better to use a function of ρ. In our case, this will be $F(\rho) = log(1+\rho)/log(2)$.

  The kernel (which is actually the spray) is a squared matrix whose side size is the rounded value of the diagonal of the original image. Therefore, the spray generation is a function dependent on the size of the image, and it will need a bigger amount of points depending on the image

(a) Spray with 400 points (b) Spray with 6400 points (c) Spray with 64000 points

Fig. 5: Sprays with different amount of points for a same sized kernel

we are working on.
Once the spray is generated, we make a convolution of the image with it in order to get an averaging kernel, and normalize it. Then, the original image is replicated so that we can apply this kernel to every pixel. This is where the approach presented is different to the one in [4], as we apply it to every pixel of the image, while they look for the maximum intensity pixels and apply a different spray for each of them.
Finally, for each pixel of the image, we apply equation (9), where *I(i,j)* is the value of the averaging kernel in that point, and the geometric and arithmetic averages correspond to the ones in an area with the centre in the pixel and the size of the diagonal of the image.

VII. Tests

All the tests performed have been a constant evolution on finding which is the best way to find the optimal semisaturation constant σ_b for equation (2). All the tested algorithms have been tested in several different HDR images, from which we would highlight the ones in Fig. 6.

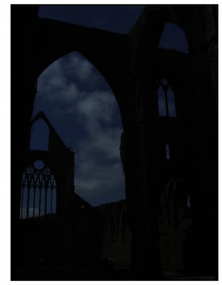
A. *Effect on color space*

Naka-Rushton equation (2) has been tested both with arithmetic mean (4) and geometric mean (5) in three different color spaces: Lab, RGB and HSV. When using the Lab color space, we only apply equation (2) to the *L* channel, this is, the *Luminance* channel. Then, we combine this modified channel with the other two original channels in order to obtain the tone mapped images.

This approach seemed rational as the luminance channel is the one changing with light directly. However, while implementing the Naka-Rushton equation on the Luminance channel, we observed that this channel gets saturated and the colors in the observed tone mapped image are not retrieved. This can be seen in Fig. 7a.

Another method we used was to not multiply the luminance channel with the Rmax, in this case the luminance channel is not saturated. However, even this method didn't give efficient results as the resulting image was too dark as shown in Fig. 7b.

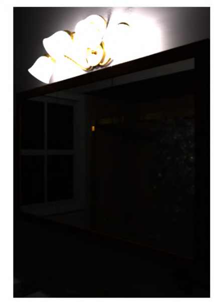
The problem with LAB color space is that its coordinates are not suitable for mapping operations. The luminance channel is elevated at a power of $1/3$ and the chromatic channels are combined to reproduce color. This makes it very complicated for tone mapping operations.



(a) tinterna.hdr

(b) memorial.hdr

(c) desk.hdr



(d) rend04_o80A.hdr

Fig. 6: Original HDR images without any tone mapping

(a) tinterna with with saturated L channel (b) tinterna with unsaturated L channel

Fig. 7: LAB colorspace

When using the RGB color space, equation (2) is applied to each of the color spaces, which are then combined to form the tone mapped image. In this case, the term R_{max} of equation (2) is ignored, otherwise it would return a white image. This colorspace gave the best results, as we could retrieve the color information. The results can be seen in Fig. 8.

With the HSV color space, the equation (2) is only applied to the *V* or *Value* channel. Then, this tone mapped channel is combined with the original *H* and *S* channels. This method gave us the color information as well but the resulting tone mapping was very saturated in color. The images can be seen in Fig. 9.
This is due to the properties of this color space. The V channel

(a) tinterna RGB with arithmetic mean (b) tinterna RGB with geometric mean

Fig. 8: RGB colorspace

corresponds to the brightness of colors. Thus, the higher the value in V channel, the brighter the corresponding color will be. Equation. (2) increases the values of the corresponding channels. In this case, this directly effects the colors in the image, causing an image saturated in color.

(a) tinterna with arithmetic mean (b) tinterna with geometric mean

Fig. 9: HSV colorspace

In the case of Lab and HSV color spaces, only one channel was considered, the one that is related to the light intensity values. As it can be seen in the images, the best and more realistic results are obtained when using the RGB color space. Therefore, from now on, all the results will be presented working in this color space.

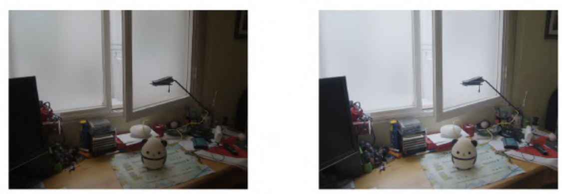
B. Global Averaging: Arithmetic and Geometric Mean

In the RGB color space, we applied equations (4) and (5) in order to obtain the arithmetic and the geometric mean of the images, respectively, in several HDR images.

As we saw in section VI-B, stated in equation (8), the arithmetic mean has a higher value than the geometric one. Therefore, the denominator in equation (2) will also be bigger, making the difference with the original image higher, and the resulting tone mapped image darker. This can be noticed experimentally by comparing the pairs of images in Fig. 10.

(a) tinterna with arithmetic mean tone mapping (b) tinterna with geometric mean tone mapping

(c) memorial with arithmetic mean tone mapping (d) memorial with geometric mean tone mapping



(e) desk with arithmetic mean tone mapping (f) desk with geometric mean tone mapping

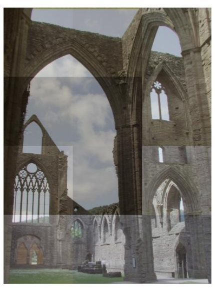
(g) rend04_o80A with arithmetic mean tone mapping (h) rend04_o80A with geometric mean tone mapping

Fig. 10: Tone mapped images with arithmetic mean and geometric mean

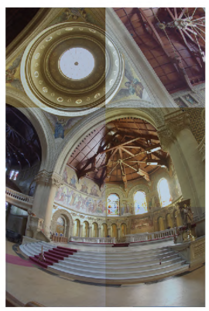
C. Local Averaging

Taking into account that applying the geometric mean returns a brighter image than applying the arithmetic mean, we considered to apply them locally depending on the value of the intensity of the pixel. This way, we would obtain a more homogeneous image, where areas of lower intensity are enlightened, and areas with higher intensity are made more visible.

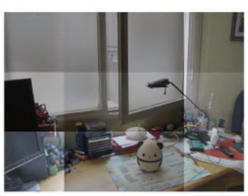
- **Kernel based local averaging:** This method was tested on a wide set of images of different types. The HDR images included scenarios such as day, night, indoor and outdoor images. This approach can be considered as the first naive approach while moving towards the local averaging. The results have two major issues.
  The first issue that occurred while applying the method was the block effect in the tone mapping. This effect occurs due to the windows used for the averaging of regions. As each region is changed to either arithmetic or geometric mean, the difference at the boundaries can be very visible. Also the value of the mean of each region is different and this difference can be seen in the resulting pixel values. Some results can be seen in Fig. 11a, 11b and 11c.



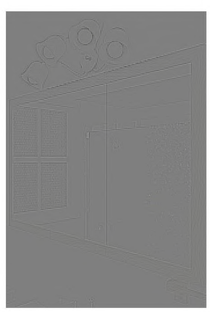
(a) tinterna



(b) memorial



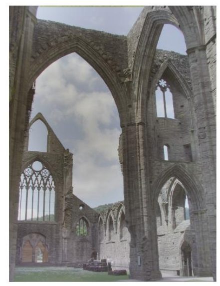
(c) desk



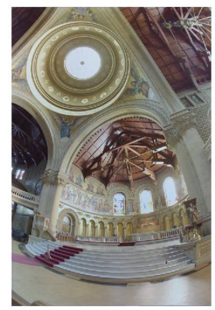
(d) $rend04_o80A$

Fig. 11: Tone mapped images with Kernel based local averaging

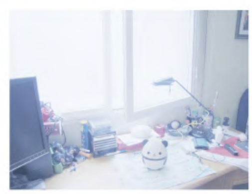
  Another problem experienced in our implementation is due to the use of greatest common divisor (GCD), to decide the window size. Using this method to find window size makes the implementation entirely dependent on the ratio of the image dimensions. If the greatest common divisor is big enough and provides a reasonable size for the windows, the result can be better. However, in case of a small GCD, the resulting image loses all the information, due to the block effect. This is visible in Fig. 11d. The dimensions of this image are 512x346 and the GCD is 2. This means that the size of the kernel is 2x2, which is too small.
  It can be concluded with the results that this approach is very naive and does not provide a proper tone mapping. For this reason, we moved on to explore other local averaging techniques.
- **Convex combination of arithmetic and geometric mean:** The method of equation (9) is very efficient to remove the block effect caused in the kernel based local averaging. By using this method, the dependency over the ratio of image dimensions is also removed. This means that this method can be used on images of any size and dimensions relation. The results of this method can be observed in Fig. 12.



(a) tinterna



(b) memorial



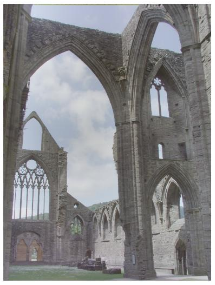
(c) desk

(d) rend04_o80A

Fig. 12: Convex combination of arithmetic and geometric mean

  The resulting images present the problem mentioned in equation (10). The equation proves that the areas of the images that are very dark can be boosted towards a very bright value. This means that the pixel value of a dark pixel would be even brighter than the one calculated using geometric mean, after the mapping. This issue can be observed from Fig. 12. The resulting image for 12a and 12b seem to be reasonably well as the original images do not have very dark areas. However, 12c and 12d seem to be very bright due to the original dark pixels in the images.
- **Linear combination of arithmetic and geometric mean:** As discussed in the previous section, the problem with the dark parts exists using the convex combination of the arithmetic and geometric mean. For this reason we de-

rive a linear combination of the arithmetic and geometric mean. This method is implemented using equation (10).



(a) tinterna

(b) memorial

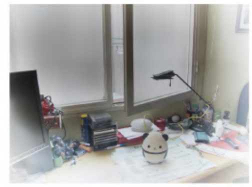
(c) desk

(d) rend04_o80A

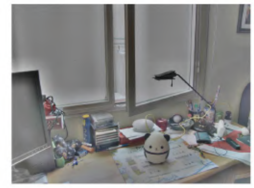
Fig. 13: Linear combination of arithmetic and geometric mean

It can be seen from Fig.13c and 13d, that the problem of very bright images disappears with the linear combination of arithmetic and geometric mean. On the other hand, the images in Fig. 13a and 13b are similar to the convex combination, which had already provided a decent result.

- **Convex combination with sprays:**
  In order to improve the results of the convex combination, we tried to apply a kernel in function (9), substituting the value of the pixel by the mean of a region obtained with the GCD of the sizes of the image.
  In order to be able to apply the kernel to the whole image, we padded it by adding zeros. This approach proved to be wrong, as the sides of the image were too bright, they looked ”burnt”, as can be seen in Fig. 14a. This can be explained because, by taking the mean of a region where most of the intensity values are zero, that value results very small and the arithmetic mean is mostly applied on it, even if it was not necessary. Then, we decided to apply a mirroring padding. In this case, the mean of the area where the padding is applied is maintained, improving significantly the result. But a new issue appears, which are the halos, which can be appreciated in Fig. 14b.
  In order to avoid the mentioned problems, we changed the way we created the kernel. Previously the kernel would just be a square matrix of ones, with the dimensions equal to the GCD of the dimensions of the original image. Now, we change the way we create that kernel and apply the *Random Spray Retinex* in [4]. This



(a) desk with zero padding



(b) desk with mirror padding

Fig. 14: Effects of zero padding and mirror padding

  approach gives more importance to the neighbouring pixels of the one where we are located, and much less importance to pixels located far away from it.
  In this approach, two parameters play a fundamental role, and have to be properly tuned. The first parameter is the size of the kernel, which will be a squared matrix whose dimension is the diagonal of the image. The other fundamental parameter is the quantity of pixels that will be considered as relevant, this is, the pixels that adopt a ”one” value in the kernel. As the relevant pixels of the spray are generated randomly, having an enough amount of them is a crucial fact in order to obtain a good result. However, it has been proved that by just ”filling” the kernel with the 1.5% of its size, properly distributed, is enough to obtain good results, as the ones presented in Fig. 15. This proper distribution can be achieved by applying the function presented when explaining equation 11. If we compare Fig. 14b and Fig. 15c, we can appreciate how the halos have disappeared, the colors are bright and the distribution of intensity is homogeneous.

VIII. Conclusions

The TRDP was broken down into two parts, The first part consisted of the generation of HDR images and this second part consisted of mapping them to make them visually compatible with display devices.

We implemented our algorithm based on the working of Human Visual System and its properties to adapt to the intensity conditions. Just like the Human Visual System, the implementation of this method can be considered as a function of the background intensity. This function can be as simple as an averaging. This implementation is done on three color spaces including LAB, RGB and HSV. We tested the LAB and HSV color space with arithmetic and geometric mean, and the RGB color space with all the developed methods. We could not retrieve the color information from the LAB color space and the information from the HSV color space was too saturated. However, the problem with the loss of colors was solved. Results from the implementation in RGB color space seemed reasonable so the remaining work was developed on the RGB color space.

An important fact observed during the implementation is that even the simple methods such as global arithmetic and geometric mean, give a reasonably good result. However, the

(a) tinterna

(b) memorial

(c) desk

(d) rend04_o80A

Fig. 15: Tone mapped images with local sprays

images recovered by using the geometric mean are brighter than the ones obtained through arithmetic mean. This is due to the fact that the arithmetic mean of a quantity is always greater than the geometric mean.

To perform tone mapping according to the local properties of the image, we implemented techniques such as kernel based averaging, convex combination of arithmetic and geometric mean, linear combination of arithmetic and geometric mean and finally, convex combination of arithmetic and geometric mean combined with sprays.

The kernel based averaging is the most naive approach having drawbacks such as block effect and image dimension ratio dependency. This method did not give desired results in any case. The next approach was a simple convex combination of the arithmetic and geometric mean. This approach helps to remove the block effects but can cause over brightness of some areas. The next method we tried is the linear combination of the arithmetic and geometric mean. This method gave reasonably good results and removed the problem of the over brightness.

Another technique that can be used to remove the block effect and overbrightness caused in kernel based averaging and convex combination of the arithmetic and geometric mean, respectively, is the random sprays. In this method, instead of pixel values we used the averaging of pixels to incorporate the information of the neighbouring pixels as well. However, just simple averaging for each pixel causes artifacts which can be removed by using kernels created through random sprays. This method gives very good results but is computationally very expensive.

IX. Future work

The work on HDR images and tone mapping is a very vast field, and can be benefited by more in the future. There are still areas open like trying other methods of tone mapping instead of the Human Visual System adaptation. Moreover, the work can be developed further on the effect equation (2) of color spaces and how to resolve the issues with each color space and other techniques of local averaging can be explored to find better results. Finally, with the actual intensity information of the scene measured physically, the algorithm can be evaluated on the basis of its similarity to the real scene, or some deep neural networks can be trained according to the ease of human eye in order to evaluate tone mapping algorithms.

Acknowledgment

We wish to thank our supervisor Edoardo Provenzi from Universite de Bordeaux, for his constant help and guidance throughout the semester. We would also like to thank Jose Maria Martinez from Universidad Autonoma de Madrid and Kristof Karacs from Pazmany Peter Katolikus Egyetem for their helpful remarks.

References

[1] ”On being undigital with digital cameras: Extending Dynamic range by combining differently exposed pictures”, Mann S., Picard R.W., Massachusets Institute of Technology, 1995.

[2] Recovering high Dynamic range radiance maps from photographs, Debevec P.E., Malik J., University of California at Berkeley, 1997

[3] ”High Dynamic Range Imaging. Acquisition, Display and Image-Based lighting”, Reinhard E., Ward G., Pattanaik S., Devebec P., Heidrich W., Myszkowski K., 2010

[4] ”Random Spray Retinex: A New Retinex Implementation to Investigate the Local Properties of the Model”, Provenzi E., Fierro M., Rizzi A., De Carli L., Gadia D., Marini D., Universit di Milano, 2007

[5] ”Light Random Sprays Retinex: Exploiting the Noisy Illumination Estimation”, Banic N., Loncaric S., University of Zagreb, 2013

[6] ”Analysis and development of tone mapping algorithms on High Dynamic Range Color images: HDR images generation”, Shams T., Herrero S., Universidad Autnoma de Madrid, 2019