**1. INTRODUCTION**

**1.1 PROBLEM STATEMENT**

Spatial resolution is not at all an issue any more in natural scenes. Realistic images are obtained either by adding third dimension or by using more and more gamut of light or color. The problem with using lighter and color is addressed by HDR imaging. It addresses the problem by capturing different images of same scene with different exposures and then composing all of them to get a single image. In contrast to traditional method we are proposing a High-Dynamic-Range imaging process from RGB to luminance-chrominance-gradient color space. The main motive is to get more efficient technique and also to avoid color-distortion generating from three different color channels if used RGB color space for HDR imaging. We introduced a camera responsive function for luminance channel so that we can find out the HDR luminance. Once the values of luminance, chrominance and gradient are extracted then tone mapping technique comes into picture which approximates the appearance of HDR Image. In many Tone mapping algorithms, there is a detail loss and higher computational time and also, they are not suitable to our method, so we are introducing Improved Bitonic Tone Mapping (IBTM).

**1.2 MOTIVATION**

Why we choose HDR Imaging? The details of the image lost due to the underexposure and overexposure. The single image does not provide the information of the objects due to some loss of data in the images. The multiple images of the same object with the different exposure can provide the needed information about the image, because different exposure of the image gives the different data. The advancement in the image acquisition technique helps to overcome the issues presented in current imaging system and make it possible to capture the scene with the higher value of dynamic range in the image from the exposure between the bright and dark area. HDR find practical use in major and critical domains like space, medical and pretty much everywhere where there is a need to show all the available useful data from different images in single image for analysis. Many approaches are proposed to capture the image with the proper dynamic range and some methods provide the ability to capture the HDR image but most HDR methods never made use of YUV space for HDR generation which by far can be able to reduce many problems arising from doing it in RGB space this is proved in our paper. The general process is to capture simultaneously with the different exposure and then irradiance spatial dimensions of the same scene. This image composition technique requires a primary calculation of camera response function, then it should be adequately tone mapped the image back to low dynamic range display. It is composed of one luminance component and two chrominance components.

**1.3 BACKGROUND**

**1.3.1 HUMAN PERCEPTION**

If we take a closer look at human perception to understand the processes on which the implemented models are based on, humans are capable of seeing a huge range of intensities, from daylight level to night luminance’s. They cope the large luminance range by a process called adaptation. A raw adaptation takes place in the first seconds, but the final adaptation takes several minutes. At this an additional distinction has to be made by either adopting to dark or to light intensities and it is also depending on the actual level of illumination. Human perception is not equal at all light intensities, because of the different sensitivities of the two different kinds of photoreceptors, the rods and cones. The human eye does a great job of compensating for the dark and light areas in a high contrast scene. When these light or dark areas reach the extremes, we need to reach for a flashlight or dark glasses, but for the average high contrast scene, our eyes do a great job of automatically compensating. Today’s digital cameras are getting close to matching this dynamic range of adjustment/compensation, but they still have a long way to go. One technique to achieve additional dynamic range in our digital images is to capture a series of photographs at differing exposure levels (exposure bracketing), then combine them into one final High Dynamic Range (HDR) image during post processing. This “combined” image would then contain the best characteristics of the series of images.

**1.3.2 COMPUTATIONAL PHOTOGRAPHY**

Computational photography refers to digital image capture and processing techniques that use digital computation instead of optical processes. Computational photography can improve the capabilities of a camera, or introduce features that were not possible at all with film-based photography, or reduce the cost or size of camera elements.

**1.3.2.1 ORDINARY FILM CAMERAS**

If you have an old-style camera, you'll know that it's useless without one vital piece of equipment: a film. A film is a long spool of flexible plastic coated with special chemicals (based on compounds of silver) that are sensitive to light. To stop light spoiling the film, it is wrapped up inside a tough, light-proof plastic cylinder the thing you put in your camera.

When you want to take a photograph with a film camera, you have to press a button. This operates a mechanism called the shutter, which makes a hole (the aperture) open briefly at the front of the camera, allowing light to enter through the lens (a thick piece of glass or plastic mounted on the front). The light causes reactions to take place in the chemicals on the film, thus storing the picture in front of you.

This isn't quite the end of the process, however. When the film is full, you have to take it to a drugstore (chemist's) to have it developed. Usually, this involves placing the film into a huge automated developing machine. The machine opens up the film container, pulls out the film, and dips it in various other chemicals to make your photos appear. This process turns the film into a series of "negative" pictures ghostly reverse versions of what you actually saw. In a negative, the black areas look light and vice-versa and all the colors look weird too because the negative stores them as their opposites. Once the machine has made the negatives, it uses them to make prints (finished versions) of your photos. If you want to take only one or two photographs, all of this can be a bit of a nuisance. Most people have found themselves wasting photographs simply to "finish off the film." Often, you have to wait several days for your film to be developed and your prints (the finished photographs) returned to you. It's no wonder that digital photography has become very popular—because it solves all these problems at a stroke**.**

**1.3.2.2 DIGITAL CAMERA**

Digital cameras take photographs using an image sensor that converts light to electronic signals. The distinguishing feature of single-lens reflex (SLR) cameras is that the optical viewfinder displays the view through the camera lens. The “single lens” in the name “single-lens reflex” is a reference to this feature: a single lens serves both for taking photographs and for the viewfinder. The “reflex” portion refers to the reflection of light. DSLR Cameras are equipped with mirrors that guide light from the lens into the viewfinder by reflecting it upward, hence the term “reflex.” The light that is reflected upward falls on the viewfinder focusing screen; after passing through the screen, it then proceeds through a pentaprism or pentamirror to the viewfinder eyepiece window. This allows the photographer to view the image from the camera lens directly in the viewfinder. When the shutter-release button is pressed all the way down, the mirror is raised and the light coming through the lens proceeds straight through to the shutter curtain. The curtain simultaneously opens to allow the light to fall on the image sensor and a photograph is taken. The shutter then closes and the mirror drops back into its original position. By linking the action of the shutter with the movement of the mirror, the light passing through the lens can be made to fall on either the viewfinder focusing screen or the image sensor.

Some newer DSLR Cameras can display the view through the lens in the monitor: this is known as “live view.” The mirror is raised during live view; blocking the light that would otherwise reach the focusing screen and preventing photographs from being framed in the view finder. Digital cameras look very much like ordinary film cameras but they work in a completely different way. When you press the button to take a photograph with a digital camera, an aperture opens at the front of the camera and light streams in through the lens. So far, it's just the same as a film camera. From this point on, however, everything is different. There is no film in a digital camera. Instead, there is a piece of electronic equipment that captures the incoming light rays and turns them into electrical signals. This light detector is one of two types, either a charge-coupled device (CCD) or a CMOS image sensor.

If you've ever looked at a television screen close up, you will have noticed that the picture is made up of millions of tiny colored dots or squares called pixels. Laptop LCD computer screens also make up their images using pixels, although they are often much too small to see. In a television or computer screen, electronic equipment switches all these colored pixels on and off very quickly. Light from the screen travels out to your eyes and your brain is fooled into see a large, moving picture.

In a digital camera, exactly the opposite happens. Light from the thing you are photographing zooms into the camera lens. This incoming "picture" hits the image sensor chip, which breaks it up into millions of pixels. The sensor measures the color and brightness of each pixel and stores it as a number. Your digital photograph is effectively an enormously long string of numbers describing the exact details of each pixel it contains. You can read more about how an image sensor produces a digital picture in our article on webcams.

**1.3.2.2.1 EXPOSURE**

“Exposure” is the act of exposing the image sensor to light. By adjusting the amount of light, you can make a photograph of a bright sunlit scene look dark, or a shot of a dark interior look bright. DSLR Cameras have auto-exposure systems that automatically produce photographs of optimal brightness. You can use this system for optimal results with both brightly-lit and poorly-lit subjects. This is referred to as “optimal exposure.”



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Reducing exposure emphasizes shadows and makes the sky a darker blue. | https://imaging.nikon.com/common/img/sp.gif | In this photograph, the camera auto-exposure system has adjusted exposure for optimal results. | https://imaging.nikon.com/common/img/sp.gif | Increasing exposure brings out details in shadows, including the road and cars. |

**Figure 1.1**: The Same Scene Photographed at Different Exposures

Here are some samples of the same scene photographed at different exposures. You may want to reduce exposure to bring out the color of the sky, or increase exposure to bring out cars and other details in shadows. The “best” exposure varies according to photographer and what details he or she considers important or wants to emphasize.

The camera meters the brightness and color of the subject and automatically adjusts exposure for optimal results.

In concrete terms, leaving the camera in charge of exposure produces optimal results with a variety of scenes. Photographers, however, may feel that brighter results would be better for some photographs and that darker results would be better for others, meaning that they may not necessarily find that the optimal exposure selected by the auto-exposure system is suitable for all photographs.

Exposure is controlled by:

1. Aperture
2. ISO
3. Shutter Speed

**1.3.2.2.2 APERTURE**

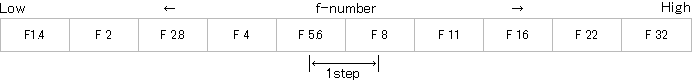
Aperture controls the brightness of the image that passes through the lens and falls on the image sensor. It is expressed as a f-number (written as “f/” followed by a number), such as f/1.4, f/2, f/2.8, /f4, f/5.6, f/8, f/11, f/16, f/22, or f/32.

Changing the f-number changes the size of the aperture, changing the amount of light that passes through the lens. The higher the f-number, the smaller the aperture and the less light that passes through the lens; the lower the f-number, the larger the aperture and the more light that passes through the lens. For example, changing the aperture from f/4 to f/5.6 halves the amount of light passing through the lens and halves the brightness of the image that falls on the image sensor.

Changing the f-number also changes the distance in front of or behind the focus point that appears to be in focus. The higher the f-number, the greater the distance in front of and behind the focus point that appears to be in focus; on the other hand, the lower the f-number, the shorter the distance in front of and behind the focus point that appears to be in focus. The distance in front of and behind the focus point that appears to be in focus is referred to as “depth of field.”

**f-numbers**

f-numbers change as shown below.



Raising the f-number one step is referred to as “stopping aperture down a step” or “stepping aperture down an f-stop.” This halves the area of the aperture (or opening), halving the brightness of the image that falls on the image sensor. Lowering the f-number by one step is referring to as “stopping aperture up a step” or “stepping aperture up an f-stop.” This doubles the area of the aperture (or opening), doubling the brightness of the image that falls on the image sensor.

**1.3.2.2.3 ISO**

In the case of digital cameras, ISO sensitivity is a measure of the camera's ability to capture light. Digital cameras convert the light that falls on the image sensor into electrical signals for processing. ISO sensitivity is raised by amplifying the signal. Doubling ISO sensitivity doubles the electrical signal, halving the amount of light that needs to fall on the image sensor to achieve optimal exposure. In other words, if ISO sensitivity is raised from ISO 100 to ISO 200 while aperture is left unchanged, the same exposure can be achieved with a shutter speed twice as fast. The same is true if ISO sensitivity is raised from ISO 200 to ISO 400.

The slow shutter speeds needed for dark interior scenes leave photographs prone to camera blur. If you raise ISO sensitivity, you can choose faster shutter speeds and reduce camera blur. This is why people say that ISO sensitivity should be raised if lighting is poor.

ISO sensitivity can be set manually by the photographer or automatically by the camera.

**No-Flash Low Light Photography**

When lighting is poor, you can use a flash to light portrait subjects. Flash units, however, have limited range. If you raise ISO sensitivity, you can optimally expose both the portrait subject and the background without using a flash at all.

|  |  |
| --- | --- |
| Photo taken with flash | Photo taken at high ISO sensitivity with flash off |
| Photo taken with flash | Photo taken at high ISO sensitivity with flash off |

**Figure 1.2:** Same photo taken twice to demonstrate ISO

**1.3.2.2.3.1 NOISE**

Raising ISO sensitivity allows faster shutter speeds, reducing blur caused by subject or camera movement. You may wonder why, if that's the case, you shouldn't simply always shoot at the highest ISO sensitivity setting, but in fact raising ISO sensitivity can introduce a type of image artifact known as “noise” into your photographs, making them seem grainy. Raising ISO sensitivity amplifies the electronic signal, which also amplifies any noise in the signal; as a result, the higher the ISO sensitivity, the more obvious the effects of noise on your photographs. The same is true of all digital cameras. We recommend that you raise ISO sensitivity only as high as needed to avoid blur.

**1.3.2.2.4 SHUTTER SPEED**

The shutter speed is the part of the camera that opens and closes to let light in and take a picture. The shutter speed is how long that shutter stays open, written in seconds or fractions of a second, like 1/200 s. or 1”, with the “symbol often used to designate an entire second. The longer the shutter stays open, the more light that is let in. But anything that moves while the shutter is open will become a blur, and if the entire camera moves while the shutter is open the whole image will be blurry—that’s why tripods are necessary for longer shutter speeds.

Shutter speed is a measurement of the time the shutter is open, shown in seconds or fractions of a second: 1 s, 1/2 s, 1/4 s … 1/250 s, 1/ 500 s, etc. The faster the shutter speed, the shorter the time the image sensor is exposed to light; the slower the shutter speed, the longer the time the image sensor is exposed to light.

If you are photographing a subject that is in motion, you will get different effects at different shutter speeds. Fast shutter speeds will “freeze” motion, while slow shutter speeds introduce blur from two sources: camera movement (camera shake) and subject. In other words, the faster the shutter speed the easier it is to photograph the subject without blur and “freeze” motion and the smaller the effects of camera shake. In contrast, slower shutter speeds are suited to suggesting the motion, such as that of flowing water or other moving subjects. Changing the shutter speed gives you control over whether to “freeze” or suggest motion.

**Blur Caused by Subject Motion at Slow Shutter Speeds**

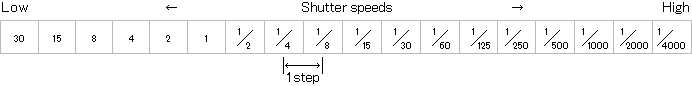
In the photograph taken at a fast shutter speed, the walker appears “frozen” in mid step. This is due to the fact that only a brief instant of the walker's motion was recorded because the shutter was only open for a short time. In the photograph taken at a slow shutter speed, the walker is blurred. This is due to the fact that the walker moved while the shutter was open.

|  |  |
| --- | --- |
| Fast shutter speeds freeze motion. | Slow shutter speeds suggest motion. |
| Fast shutter speeds freeze motion. | Slow shutter speeds suggest motion. |

**Figure 1.3:** images taken to demonstrate shutter speeds

Shutter Speed Values

Shutter speeds change as shown below.



Choosing a shutter speed one step faster than the current shutter speed (by, for example, changing shutter speed from 1/60 s to 1/125 s) is referred to as “increasing shutter speed by one step” and halves the amount of time the shutter is open. Choosing a shutter speed one step slower than the current shutter speed (for example, by changing shutter speed from 1/125 s to 1/60 s) is referred to as “slowing shutter speed by one step” and doubles the amount of time the shutter is open.

|  |  |
| --- | --- |
| Fast shutter speed | Slow shutter speed |
| Fast shutter speed | Slow shutter speed |

**Figure 1.4:**  images taken to show blur formed by slow shutter speed

**1.3.2.2.5 COMBINING APERTURE AND SHUTTER SPEED**

The combination of aperture (f-number) and shutter speed determines exposure (another important factor in determining exposure is ISO sensitivity, but in the discussion that follows we will assume that ISO sensitivity is fixed). Choosing higher f-numbers correspondingly darkens the image that falls on the image sensor, but you can still achieve optimal exposure if you slow shutter speed in proportion. On the other hand, you can also achieve optimal exposure by choosing a lower f-number and a faster shutter speed. In other words, there are many combinations of aperture and shutter speed that will produce the same exposure. For example, suppose the subject will be correctly exposed at an aperture of f/4 and a shutter speed of 1/250 s. The correct exposure will then also be achieved if you increase the f-number to f/5.6 while slowing shutter speed to 1/125 s. The same is true at f/8 and 1/60 s and at f/11 and 1/30 s.

Combinations of aperture and shutter speed that will produce the same exposure as f/4, 1/250 s

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Aperture | F16 | F11 | F8 | F5.6 | F4 | F2.8 | F2 | F1.4 |
| Shutter speed(seconds) | 1/15 | 1/30 | 1/60 | 1/125 | 1/250 | 1/500 | 1/1000 | 1/2000 |

**Table 1.1:** Aperture and shutter speed combinations

If you always adjust shutter speed to match any changes in aperture, you can achieve correct exposure at any aperture or shutter speed. Note, however, that changing aperture also changes depth of field, while changing shutter speed alters the appearance of moving objects. In other words, you can also adjust aperture for depth of field or shutter speed to produce the effect of motion.

**Camera Blur and Motion Blur**

If the camera or subject moves while the shutter is open, the picture will be blurred. Blur caused by subject movement is referred to as “subject blur” or “motion blur”; blur caused by camera movement (“camera shake”) is referred to as “camera blur.” The results in both cases are similar, but whereas blur caused by subject movement is generally regarded as a legitimate way of expressing motion in photographs, blur caused by camera shake is frequently seen as a flaw. While camera blur does not necessarily render a photograph a failure, caution should be observed to avoid unintentional camera blur. The main subject is in both cases blurred, but the results are distinct from blur caused by the subject being out of focus (focus blur).

|  |  |
| --- | --- |
| **Camera blur** | **Motion blur** |
| Camera blur | Motion blur |
| The camera moved while the shutter was open, producing blur. | The main subject moved in the wind while the shutter was open and is blurred; the surrounding flowers and leaves, which were at rest while the shutter was open, are not |
| **Out-of-focus shot (focus blur)** | **Intended result** |
| Out-of-focus shot (focus blur) | Intended result |
| The camera is focused not on the flower in the centre but on a flower further back. | The central flower is in focus. |

**Figure 1.5:** Camera blur and motion blur

**1.3.2.2.6 ASPECT RATIO**

If you’ve ever printed images before, you’ve probably noticed that an 8 x 10 usually crops from the original image. That’s due to aspect ratio. Aspect ratio is simply the ratio of the height to width. An 8 x 10 has an equal aspect ratio to a 4 x 5, but a 4 x 7 image is a bit wider. You can change the aspect ratio in your camera if you know how you’d like to print your image, or you can crop your photo when you edit it to the right ratio.

**1.3.2.2.7 BURST MODE**

You can take photos one at a time. Or, you can turn the burst mode on and the camera will continue snapping photos as long as you hold the button down, or until the buffer is full (which is a fancy way of saying the camera can’t process anymore). Burst speeds differ based on what camera or film camera you own, some are faster than others. Just how fast is written in “fps” or frames (pictures) per second. This will give you a wide selection of which close-up you’ll ultimately select of your dog!

**1.3.2.2.8 DEPTH OF FIELD**

Depth of field is a photography term that refers to how much of the image is in focus. The camera will focus on one distance, but there’s a range of distance in front and behind that point that stays sharp—that’s depth of field. Portraits often have a soft, unfocused background—this is a shallow depth of field. Landscapes, on the other hand, often have more of the image in focus—this is a large depth of field, with a big range of distance that stays sharp.

**1.3.2.2.9 FOCAL LENGTH**

The focal length describes the distance in millimeters between the lens and the image it forms on the film. It informs the angle of view (how much of what is being shot will be captured) and the magnification (how large things will appear). Essentially, the focal length is how ‘zoomed in’ your images will appear. For example, a Canon (or Nikon or Olympus) 35mm lens will create images that appear more ‘zoomed in’ than a Canon 18mm.

**1.3.2.2.10 FOCUS**

When your eyes focus on an object that’s close to you, the objects far away will appear blurry. The common photography term “focus” has the same meaning. Something that is in focus is sharp, while an object that is out-of-focus isn’t sharp. Different focus areas determine if the camera is focusing on multiple points or one user-selected point.

.

**1.3.2.2.11 WHITE BALANCE**

White balance is used to adjust colors to match the color of the light source so that white objects appear white. Subjects may be lit by a number of different light sources, including sunlight, incandescent bulbs, and fluorescent lighting. Although to the naked eye all these different light sources may appear colourless, in fact they emit light of different colours. The image sensor in a digital camera will reproduce these color differences just as they are, with the result that without additional processing the color of the photograph would appear to change according to the light source. Auto white balance automatically processes the image to remove unwanted color casts by, for example, making photographs taken under incandescent bulbs more blue to correct the reddish cast of this type of lighting. Normally, auto white balance will produce the desired results without the photographer having to worry about the type of lighting. If auto white balance does not produce the desired results, the photographer can choose from a number of fixed white balance options according to the weather or the light source. The photographer can also choose a setting for direct sunlight or incandescent lighting to introduce a deliberate red or blue cast according to their creative intent.

|  |  |  |
| --- | --- | --- |
| Auto white balance | “Direct sunlight”selected for a red cast | “Incandescent”selected for a blue cast |
| Auto white balance | “Direct sunlight” selected for a red cast | “Incandescent” selected for a blue cast |

**Figure 1.6:** same images with different white balance settings

**1.3.3 IMAGE PROCESSING TERMINOLOGY**

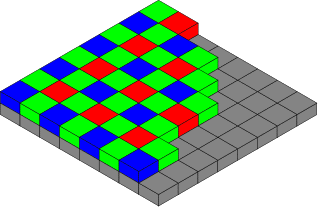
**1.3.3.1 DIGITAL IMAGE PROCESSING**

An image may be defined as a two-dimensional function, f (x, y), where x and y are spatial (plane) coordinates, and the amplitude off at any pair of coordinates (x, y) is called the intensity or gray level of the image at that point. When x, y, and the amplitude values of f are all finite, discrete quantities, we call the image a digital image. The field of digital image processing refers to processing digital images by means of a digital computer. Note that a digital image is composed of a finite number of elements, each of which has a particular location and value. These elements are referred to as picture elements, image elements and pixels. Pixel is the term most widely used to denote the elements of a digital image.

**1.3.3.2 IMAGES**

**Photo:**  
A photo is simply light reflected by an object and captured by a camera. A grayscale camera records the intensity of reflected light. A colour camera records the light reflected at three points of the visible spectrum: Red (650 nm), Green (510 nm) and Blue (475 nm).

**Colour:**  
Colour is a fascinating subject. As kids, we were taught that the Rainbow has all the colours in the world. That is absolutely wrong! “A Rainbow does not contain all colours humans can perceive!”. In fact, the colour pink is not present in the rainbow! It does not correspond to any wavelength of light. Pink is a mixture of Blue and Red color is, therefore a mechanism of the human eye by which it perceives a mixture of lights of different wavelengths.



**Figure 1.7:** Color Filter Array

How is a color recorded by a digital camera?

Inside your digital camera, there is a sensor that is sensitive to visible light (photons). The sensor has many photo sites or pixels arranged in a rectangular grid. These photo sites produce an electrical charge directly proportional to the amount of light (photons) it receives.

Depending on the scene we are recording, different pixels of the sensor are exposed to different amounts of light and we will see a pattern of electrical charge. The amount of electrical charge is converted to a number between 0 and 255. This corresponds to 8-bits or 256 values.

This rectangular matrix of numbers between 0 and 255 is a digital grayscale image.

Color sensors are much more complicated. Every pixel in the sensor records only one color – Red, Green or Blue. The pattern shown above is called the Bayer Pattern. Notice there are twice as many green pixels compared to blue and red pixels because the human eye is much more sensitive to the green light compared to the red or the blue light. The two missing channels at every pixel are calculated by interpolating the values from neighbouring pixels. This process is called demosiacing.

The story does not end there. Our eyes are not linear sensors. If you are staring at a dimmed light source and you double the intensity of light, the perceived change in intensity will be much less than 2x. To mimic this behaviour, cameras apply a nonlinear transform to the recorded image intensity. This transformation is often approximated by a single parameter function called the gamma. Can you get access to the linear image (i.e. the image without gamma correction) the sensor records? Yes, in many SLR cameras you have the option to record the image in the RAW format which is nothing but the linear image with the Bayer mosaic intact. RAW images also contain more than 8 bits per channels.

**How are images stored on disk?**

As mentioned in the previous section, once the image is read out from the sensor, it is demosaiced and usually converted to an RGB image. Usually, this array of RGB pixels is compressed to JPG or PNG formats before storing on disk. Most image formats, like JPG, have two parts:

* Image Header: This part contains image metadata like the width and height of the image, number of channels, the color profile, the number of bits per pixel and so on and so forth.
* Data: This part contains compressed RGB values.

**1.3.3.2.1 COMPRESSION OF IMAGES**

Imagine for a moment that you're a CCD or CMOS image sensing chip. Look out of a window and try to figure out how you would store details of the view you can see. First, you'd have to divide the image into a grid of squares. So, you'd need to draw an imaginary grid on top of the window. Next, you'd have to measure the color and brightness of each pixel in the grid. Finally, you'd have to write all these measurements down as numbers. If you measured the color and brightness for six million pixels and wrote both down both things as numbers, you'd end up with a string of millions of numbers—just to store one photograph! This is why high-quality digital images often make enormous files on your computer. Each one can be several megabytes (millions of characters) in size.

To get around this, digital cameras, computers, and other digital gadgets use a technique called compression. Compression is a mathematical trick that involves squeezing digital photos so they can be stored with fewer numbers and less memory. One popular form of compression is called JPG (pronounced J-PEG, which stands for Joint Photographic Experts Group, after the scientists and mathematicians who thought up the idea). JPG is known as a "lossy" compression because, when photographs are squeezed this way, some information is lost and can never be restored. High-resolution JPGs use lots of memory space and look very clear; low resolution JPGs use much less space and look more blurred. You can find out more about compression in our article on MP3 players.

Most digital cameras have settings that let you take pictures at higher or lower resolutions. If you select high-resolution, the camera can store fewer images on its memory card—but they are much better quality. Opt for low-resolution and you will get more images, but the quality won't be as good. Low-resolution images are stored with greater compression.

**NEF (RAW)**

NEF (RAW) files record raw data from the camera image sensor. Camera settings are saved separately from the raw data, allowing settings such as exposure and white balance to be changed after shooting. The same image can be edited in a variety of different ways without affecting image quality.

**JPEG**

JPEG images are compressed using an image-processing algorithm that reduces file size partly by reducing the amount of information the images contain. The amount of compression can be selected from “Fine”, “Normal”, and “Basic” for compression ratios of approximately 1 : 4, 1 : 8, and 1 : 16, respectively. The lower the compression ratio, the better the quality and the larger the file size. The higher the ratio, the smaller the file size but the greater the loss of information.

**TIFF**

TIFF is an uncompressed format and causes no drop-in image quality. It is also supported by a wide variety of image software. TIFF files are however comparatively large.

“Image quality” refers to the file type and compression ratio used when images are saved, while “image size” (measured in pixels) determines the physical dimensions of the image.

**Image Quality and File Type (NEF/RAW, JPEG, and TIFF)**

Depending on the option selected for image quality, images will be recorded in NEF (RAW), JPEG, or TIFF formats (the options available vary with the model of camera).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Image quality | | File name | Viewable in | Direct print/DPE | Compression (approximate compression ratio) | File size |
| NEF (RAW) | | DSC\_xxxx.NEF (NEF = Nikon Electronic For-mat) | Software that supports NEF (RAW) format images | Not support-ed | Compressed. High-end cameras also support uncom-pressed NEF/RAW. | Large |
| JPEG | Fine | DSC\_xxxx.JPG | A variety of general-purpose software | Supported | Compressed (1 : 4) | Large (for JPEG images) |
| Normal | Compressed (1 : 8) | Medium (for JPEG images) |
| Basic | Compressed (1 : 16) | Small (for JPEG images) |
| TIFF | | DSC\_xxxx.TIF | General-purpose imaging software | Not support-ed | Not compressed | Very large |

**Table 1.2:** Image formats and types

**Image Editing**

|  |  |  |
| --- | --- | --- |
| Image quality | Edit using | Suited to later editing? |
| NEF (RAW) | ViewNX 2 or Capture NX 2 | Yes |
| JPEG | A variety of general-purpose software | No |
| TIFF | General-purpose imaging software | Yes |

**Table 1.3:** Editing tools for different image formats

**1.3.3.3 OpenCV:**

OpenCV is the most popular Computer Vision library in the world with an estimated 14 million downloads. The following qualities make it an excellent library of choice for building commercial Computer Vision applications.

* Highly optimized: OpenCV is written in C/C++ with the goal of building real-time applications. It is highly optimized when compiled with the appropriate options and can utilize multiple cores on your machine. It is also capable of utilizing the heterogenous computing resources (e.g. a GPU) when compiled with OpenCL support.
* Open sourced under the BSD license: It is open source and licensed under the very permissive BSD license. This means you can use it to build commercial applications and do not need to open source your own code. However, there are parts of OpenCV (e.g the opencv\_contrib module) that may or may not be under the BSD license.
* Language bindings: It is written in C/C++ with bindings for other languages including Python and Java.
* Portability: It supports Linux, Mac, Windows, iOS and Android operating systems.

How are images represented in memory in OpenCV?

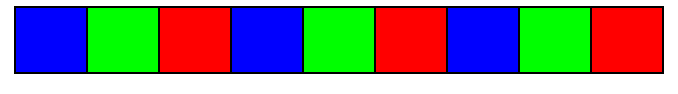
When we read an image using OpenCV, it first decompresses the image, and stores all images in a standard format.

**OpenCV Images in C++**

In the C++ version, images are instances of the Mat class. The Mat class has two parts

* Image Header: This part contains image metadata like the width and height of the image, number of channels, the number of bits per pixel and so on and so forth.
* Data: This part is an array of uncompressed BGR values in a row major format. Why not RGB? OpenCV stores images in BGR format for historical reasons.

The first three bytes in this data array correspond to the blue, green and red channels in the top left corner of the image. The next 3 bytes correspond to the pixel in the second column and first row of the image. All the pixels in one row of an image are stored as a continuous block in memory. The image below shows 3 pixels.

 **Figure 1.8:** pattern of storage of Bayer filter values in image data

**1.3.3.4 WORKING WITH IMAGES ON MATLAB**

**Image Data:**

The basic MATLAB® data structure is the *array*, an ordered set of real or complex elements. An array is naturally suited to the representation of *images*, real-valued, ordered sets of color or intensity data. (An array is suited for complex-valued images.)

In the MATLAB workspace, most images are represented as two-dimensional arrays (matrices), in which each element of the matrix corresponds to a single pixel in the displayed image. For example, an image composed of 200 rows and 300 columns of different coloured dots stored as a 200-by-300 matrix. Some images, such as RGB, require a three-dimensional array, where the first plane in the third dimension represents the red pixel intensities, the second plane represents the green pixel intensities, and the third plane represents the blue pixel intensities.

This convention makes working with graphics file format images similar to working with any other type of matrix data. For example, you can select a single pixel from an image matrix using normal matrix subscripting:

I(2,15)

This command returns the value of the pixel at row 2, column 15 of the image I.

The following sections describe the different data and image types, and give details about how to read, write, work with, and display graphics images; how to alter the display properties and aspect ratio of an image during display; how to print an image; and how to convert the data type or graphics format of an image.

**Data Types**

MATLAB math supports three different numeric classes for image display:

* double-precision floating-point (double)
* 16-bit unsigned integer (uint16)
* 8-bit unsigned integer (uint8)

The image display commands interpret data values differently depending on the numeric class the data is stored in. 8-Bit and 16-Bit Images includes details on the inner workings of the storage for 8- and 16-bit images.

By default, most data occupy arrays of class double. The data in these arrays is stored as double-precision (64-bit) floating-point numbers. All MATLAB functions and capabilities work with these arrays.

For images stored in one of the graphics file formats supported by MATLAB functions, however, this data representation is not always ideal. The number of pixels in such an image can be very large; for example, a 1000-by-1000 image has a million pixels. Since at least one array element represents each pixel, this image requires about 8 megabytes of memory if it is stored as class double.

To reduce memory requirements, you can store image data in arrays of class uint8 and uint16. The data in these arrays is stored as 8-bit or 16-bit unsigned integers. These arrays require one-eighth or one-fourth as much memory as data in double arrays.

**Bit Depth**

MATLAB input functions read the most commonly used bit depths (bits per pixel) of any of the supported graphics file formats. When the data is in memory, it can be stored as uint8, uint16, or double. For details on which bit depths are appropriate for each supported format, see imread and imwrite.

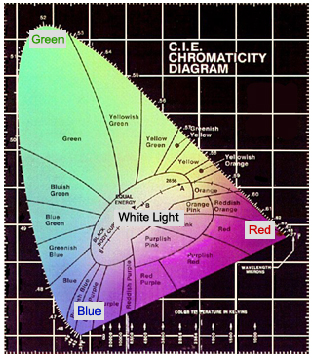
**Supported Image Formats**

MATLAB commands read, write, and display several types of graphics file formats for images. As with MATLAB generated images, once a graphics file format image is displayed, it becomes an image object. MATLAB supports the following graphics file formats, along with others:

* BMP (Microsoft® Windows® Bitmap)
* GIF (Graphics Interchange Files)
* HDF (Hierarchical Data Format)
* JPEG (Joint Photographic Experts Group)
* PCX (Paintbrush)
* PNG (Portable Network Graphics)
* TIFF (Tagged Image File Format)
* XWD (X Window Dump)

**1.3.3.5 COLOR SPACES**

A range of colors can be created by the primary colors of pigment and these colors then define a specific color space. **Color space**, also known as the color model (or color system), is an abstract mathematical model which simply describes the range of colors as tuples of numbers, typically as 3 or 4 values or color components (e.g. **RGB**). Basically speaking, color space is an elaboration of the coordinate system and sub-space. Each color in the system is represented by a single dot.

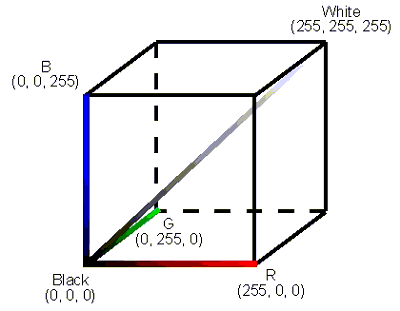


**Figure 1.9:** example color space 2-D diagram

A color space is a useful method for users to understand the color capabilities of a particular digital device or file. It represents what a camera can see, a monitor can display or a printer can print, and etc.

**1.3.3.5.1 RGB Color Space**

RGB (R=Red, G=Green, B=Blue) is a kind of color space which uses red, green and blue to elaborate color model. An RGB color space can be simply interpreted as "all possible colors" which can be made from three colors for red, green and blue. In such conception, each pixel of an image is assigned a range of 0 to 255 intensity values of RGB components. That is to say, using only these three colors, there can be 16,777,216 colors on the screen by different mixing ratios.



**Figure 1.10:** RGB color space 3-d view

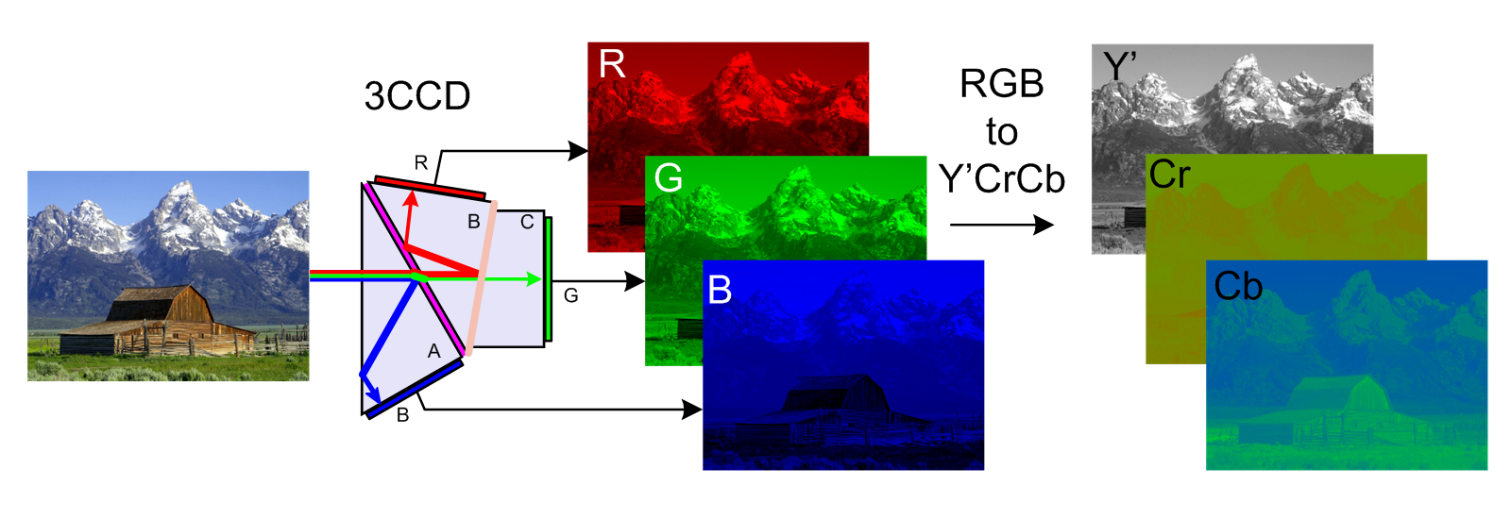
Because the human eye only has color sensitive receptors for red, green and blue, it is theoretically possible to decompose every visible color into combinations of these three “primary colors.” Color monitors, for instance, can display millions of colors simply by mixing different intensities of red, green and blue. It is most common to place the range of intensity for each color on a scale from 0 to 255 (one byte). The range of intensity is also known as the “color depth”. The possibilities for mixing the three primary colors together can be represented as a three-dimensional coordinate plane with the values for R (red), G (green) and B (blue) on each axis. This coordinate plane yields a cube called the RGB color space. If all three color channels have a value of zero, it means that no light is emitted and the resulting color is black (on a monitor, for example, it cannot be blacker than the surface of the monitor producing 0 light). If all three color channels are set to their maximum values (255 at a one-byte color depth), the resulting color is white. This type of color mixing is also called “additive color mixing”.

If you draw a diagonal from the black (0, 0, 0) origin point of the color cube to the white (255, 255, 255) point, you will get a line where each point on the line has identical R,G and B values. The result of having the same value for all three color channels is the color gray. The only thing that changes as you move along this diagonal is the intensity of the shade of gray as you go from black to white.

**1.3.3.5.2 YCBCR**

YCbCr, Y′CbCr, or Y Pb/Cb Pr/Cr, also written as YCBCR or Y'CBCR, is a family of color spaces used as a part of the color image pipeline in video and digital photography systems. Y′ is the luma component and CB and CR are the blue-difference and red-difference chroma components. Y′ (with prime) is distinguished from Y, which is luminance, meaning that light intensity is nonlinearly encoded based on gamma corrected RGB primaries.

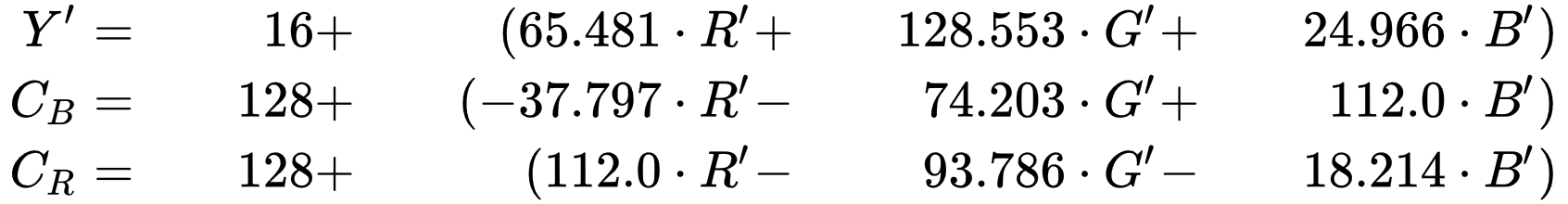
Y′CbCr color spaces are defined by a mathematical coordinate transformation from an associated RGB color space. If the underlying RGB color space is absolute, the Y′CbCr color space is an absolute color space as well; conversely, if the RGB space is ill-defined, so is Y′CbCr.

 **Figure 1.11:** RGB to YCrCb transformation

Analog YPbPr from analog R'G'B' is derived as follows:

{\displaystyle {\begin{aligned}Y’&=&0.299\cdot R’&+&0.587\cdot G’&+&0.114\cdot B’\\P\_{B}&=-&0.168736\cdot R’&-&0.331264\cdot G’&+&0.5\cdot B’\\P\_{R}&=&0.5\cdot R’&-&0.418688\cdot G’&-&0.081312\cdot B’\end{aligned}}}

Digital Y′CbCr (8 bits per sample) is derived from analog R'G'B' as follows:



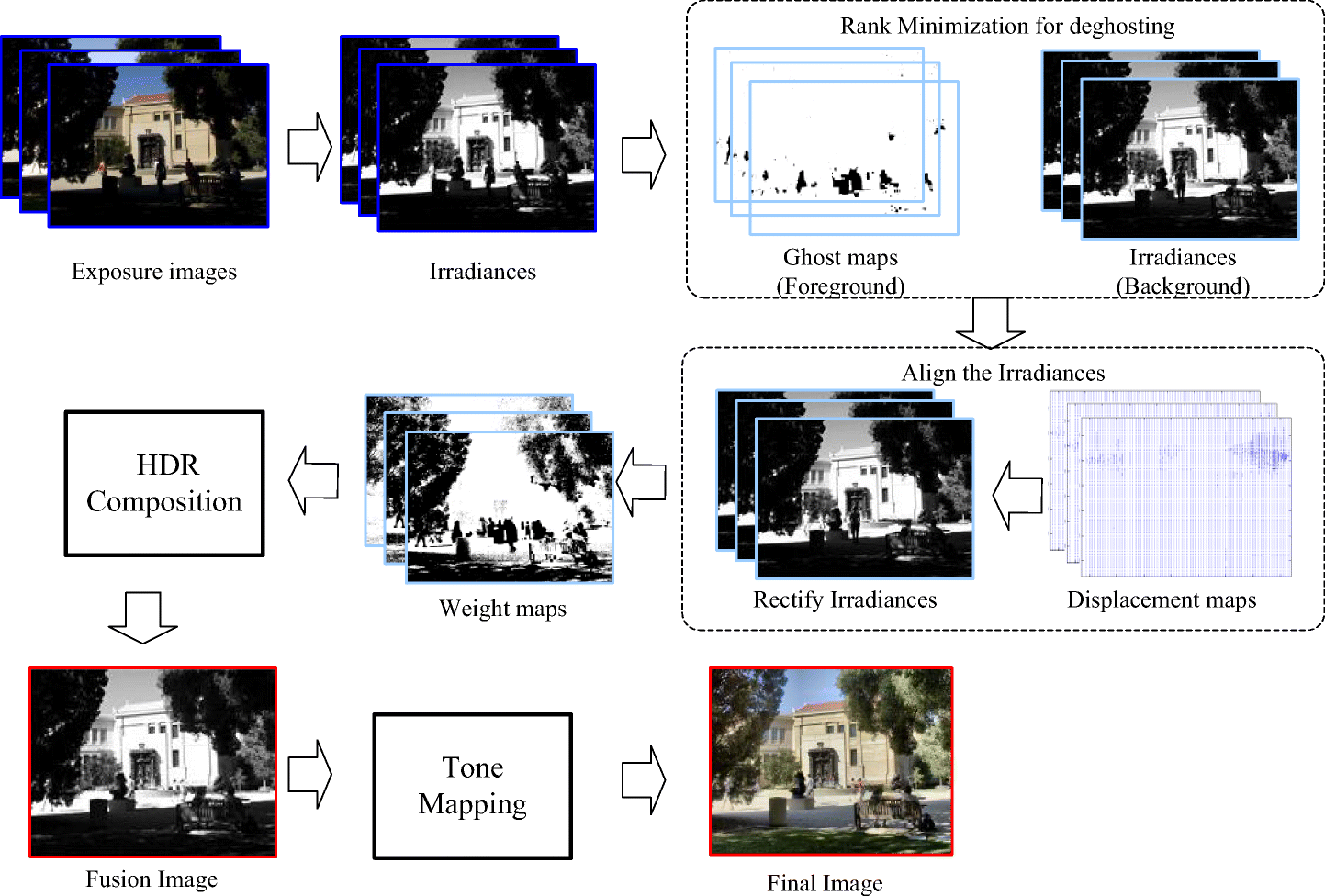
**1.4 ORGANISATION OF THE BOOK**

The rest of the book is organised as follows. In chapter 2 literature survey is being put forth it explains briefly about various other’s work and their methodology that is being used to in HDR technology. In chapter 3 we deal with already Existing methods in HDR imaging and tone mapping and how they work and some of their drawbacks, second part of chapter deals with our proposed system i.e..., its architecture at various levels, its working pipeline and also descriptive illustration of the Algorithms that make up our method. Chapter 4 deals with experimental results and comparison with other methods that are described in chapter 3 and chapter 5 deals with conclusion and future scope of the project

**2. LITERATURE SURV****EY**

The latest research papers on the HDR imaging techniques taken for the literature review. This gives the current techniques in the HDR imaging and also their advantages and limitations for the respective research.

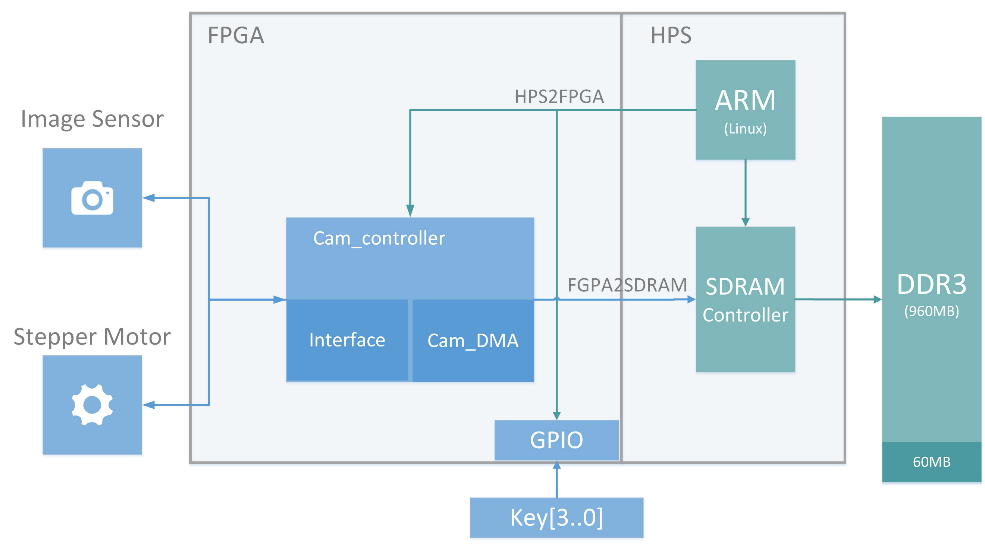
**2.1 SPARSE REPRESENTATION FRAMEWORK**

Qingsen Yan, *et al*. utilized the sparse representation framework for the HDR image synthesis algorithm with ghost free. The ghost artifacts created due to the adjacent Low Dynamic range (LDR) image and the moving object sparsity. The problem formulated into the two process: detection of moving object and ghost free HDR generation. The proposed method has the higher performance than the existing methods based on the textures and colors. The overall performance of this technique was better and in the few databases, the value of the PSNR was less than those other methods.

**Figure 2.1:** Tone Mapping Pipeline

**2.2 CALIBRATED EMBEDDED SYSTEM**

Yuije Wu, *et al*. [3] established the calibrated embedded system for quasi real-time lighting for HDR sky monitoring. The direction of the device was set in direct of a test module with unilateral-facades to calculate the distribution of the luminance in sky and the ground dome. The measurement of the luminance distribution made for the sun, landscape and ground dome, and on-board illuminance processed in the device. This proposed method and the Perez all-weather sky model was compared and this method attained the more reliable, which had the 10%-25% more accuracy due to HDR imaging and luminance mapping in transient lighting computation of horizontal illuminance. This can possibly help to improve the system that rejection of sun’s component in the luminance map.



**Figure 2.2:** Embedded system Architecture

**2.3 EXPOSURE CONDITION ANALYSIS**

Yang Song, *et al*. [4] proposed exposure condition analysis method based on the quality evaluation method for the tone-mapped HDR images. First, local exposure property analyzed for the purpose-designed HDR exposure segmentation model that used to separate image. The two new quality features such as abnormal exposure ration and exposure residual energy extracted and it is low complex. The color-based feature was also extracted and these features in the different exposure region. The quality evaluation model implemented by regression training. The ability of the model to predict the quality of tone-mapped HDR images showed in their experiment. The Pearson linear correlation coefficients are higher than 0.88, this technique have the high consistent with human visual perception.

**2.4 HDR BLEEDING ALGORITHM**

Hyuk-Ju Kwon, *et al.* [5]proposed the new method in the HDR bleeding algorithm, which used only the dual-exposure image. The least squares method used in the proposed algorithm and it also included the spatial and intensity weighting functions. The error point was reduced and improved the Camera Response Function using the weighting function. In addition, a constraint was added to correct the white balance in the brightness level. The result showed that the proposed algorithm outperformed the existing method. This method was not evaluated in the real time function and this gives the efficiency of the system.

**2.5 USING A MODIFIED GRADIENT FUNTION IN TONE MAPPING**

David Gommelet, et al. [1] analyzed the problem of tone mapping operator for rate distortion optimized backward compatible compression of HDR images. The distortion and rate are expressed as a function of the gradient in the HDR image. The proposed rate and distortion model are based on the function of the gradient in the HDR image. The experimental result shows that the proposed operator achieves optimal value for rate-distortion performance in global tone mapping. The second tone mapping operator has trade-off performance of the distortion rate and quality preservation. The quality of the image should be preserved and evaluated on various dataset.

**2.6 HIGHER ORDER RECONSTRUCTION TECHNIQUE**

Alessandro Artusi, et al. [2] proposed High Order Reconstruction (HOR) technique to preserve the edge in multi-scale edge aware tone mapping. The edge-aware technique increases the quality by smoothing an input images, while keeping its edges intact. The proposed HOR technique focus on various factors of the performance such as altering the image structure due to changes in contrast; removes artifacts around edges; as well as reducing computational complexity in terms of implementation and associated computational costs. The proposed technique aims to reduce the changes in the image structure by using an edge-stop mechanism, whose computational cost is compared to the state-of-art method. In the 18th evaluated image shows that the proposed HOR has the higher value for the loss of contrast. This technique is not much suitable to apply for the images with high contrast (blue) due to low amplification.

**2.7 DEBEVEC’S EXTRACTION OF CAMERA RESPONSIVE FUNCTION**

The construction of radiance map in many HDR techniques is based on the idea of [Debevec and Malik 1997] paper [6]. This process mainly consists of two steps: The first step is to identify the response curve 'g' that maps the pixel values to the log of exposure values for three separate color channels [see the code 'g\_curve.m']. Once the curve 'g' is available, the second step is to map the observed pixel value and exposure time in the exposure image set to the radiance for each pixel using a weighting function 'w’. Before the two main steps, a point-selecting step is performed to automatically randomly select sufficient number of pixels that are evenly distributed from range 0 to 255.

**2.9 THE BITONIC FILTER**

Graham treece [7] has proposed a new ﬁlter which has better edge and detail preserving properties than a median, noise reduction capability very similar to a Gaussian, and is applicable to many signal and noise types. It is built on a deﬁnition of signal as bitonic, i.e. containing only one local maxima or minima within the ﬁlter range. This deﬁnition is based on data ranking rather than value, hence the bitonic ﬁlter is non-linear, comprising a combination of morphological and linear operators. It has no data-level-sensitive parameters and can locally adapt to the signal and noise levels in an image, precisely preserving both smooth and discontinuous signals of any level when there is no noise, but also reducing noise in other areas without creating additional artefactual noise.

**3.METHODOLOGY**

**3.1 EXISTING SYSTEMS**There are many existing methods for HDR imaging and tone mapping. Almost all of them have been developed generically for the RGB color space. Luminance chrominance color space representations have been neglected in these methods. However, HDR imaging techniques working in luminance chrominance seem more meaningful and preferable for the following reasons.

1. Most image compression techniques store images in some Luminance chrominance space so, it is more efﬁcient to compose the HDR image directly in the same color space.
2. Any HDR technique operating in RGB space requires post-composition white balancing since the three-color channels undergo parallel transformations. For the sake of hue preservation and better compression, it is beneﬁcial to opt for a luminance-chrominance space, even if the input data is in RGB, e.g. uncompressed TIFF.
3. The luminance channel, being a weighted average of the R, G, and B channels, enjoys a better signal-to-noise ratio (SNR), which is crucial if the HDR imaging process takes place in noisy conditions.

Generally, the low dynamic range (LDR) images are merged to produce an HDR image using some merging techniques first before application of any tone mapping algorithms. The following are some of the merging techniques

1. Debevec
2. Robertson
3. Mertenes Fusion

The following are some of the famous Tone Mapping methods already defined

1. Reinhard
2. Durand
3. Drago

**3.1.1 REINHARD’S HDR METHOD**

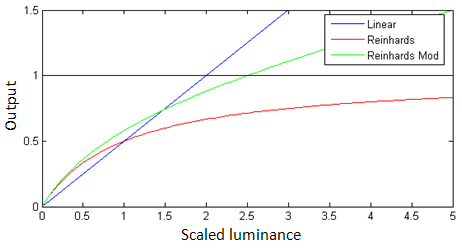
The Reinhard’s operator is named after Erik Reinhard who published this tone mapper in his work. It uses a nonlinear mapping to display the whole dynamic range of an image. The first step is to scale the HDR with a key and the logarithmic average scene luminance to LDR using Equation Further it makes a compression of high luminance values with this formula.

Color = Lscaled / 1+Lscaled

This operator scales high luminance values by 1L and low values by almost 1, so it maps all values in the range [0,1].



**Figure 3.1:** Comparison of Reinhard’s and modified Reinhard’s operator



**Figure 3.2:** Comparison of operators

High Dynamic Range images can be created using sequence of images captured at different exposure times. It is very difficult to capture the full dynamic range images by using the modern digital camera. Combining different low dynamic range (LDR) images with different exposure time the HDR image can be created. In 1997 the algorithm was developed which can create HDR radiance map. Algorithm can store the HDR image in RGB form 

**Figure 3.3:** Low Dynamic Range photograph with short, medium and long exposure time

The above figure represents three shots taken under different exposure values of the camera. The first one is underexposed image which shows contrast image similarly, the third one is overexposed image which shows brighten image. The medium exposure value image is not able to present all luminary information of image. To overcome from this problem the tone mapping method was invented. From all 3 LDR images the HDR image can be generated using tone mapping method. The block diagram of HDR image processing system is as shown in Fig. in which 7 blocks are present different functions. The multi exposure digital camera can take N number of images, by adding those images pixel values individually and RGB pixel values can taken separately the HDR image found, by applying tone mapping the HDR images are converted into LDR images, which can be able to display on device. The block diagram shows very simple method of HDR image generation.

Tone mapping method is used to reproduce the image and provides mapping between luminance of original scene to output device display values. The problem with standard displays is that they are unable to display High Dynamic Range images. To solve this problem the tone mapping technique was invented, it can display the maximum luminance image on standard display without compromising quality of image by converting High Dynamic Range image into Low Dynamic Range image. The classification of tone mapping operators in four classes: global, local, gradient domain and frequency domain operator.

**3.1.2 DURAND’S HDR METHOD**

The idea of Tone mapping of Durand [Durand 2002] is based on a bilateral filtering on a separable low-frequency layer. After merging of LDR images into HDR radiance map, the resulting radiance map is need to compressed by a tone mapping operator so as to fit into the displayable range (0 to 255) or (0 to 1) if it is decimal. In general, there are two class of tone mapping methods. One method from global tone mapping class is to calculate the pixel value for each radiance value L to be 'L / (1+L) However, the result of global operator is not satisfactory since the details of both brighter and darker regions cannot be distinguished very well at the same time.

The method contains the following steps:

* 1. separate the intensity image from the color components and the intensity is compressed into the log domain.
  2. divide the intensity image into the detailed and large-scale components. To do this, one should first obtain the large-scale layer by blurring the intensity image using an edge-preserving bilateral filter which takes both spatial and intensity range into consideration at the same time. Then, the detailed layer can be recovered by subtracting the blurred large-scale layer from the intensity image.
  3. the large-scale layer is compressed in order to reduce the contrasts while keeping the other components intact.
  4. the compressed large-scale layer, the detail layer and the color layer are recombined to form the HDR image.
  5. a gamma correction is used to a just the final HDR image.

**3.1.3 DRAGO’S HDR METHOD**

The Drago tone mapping operator is an implementation of the "Adaptive Logarithmic Mapping For Displaying High Contrast Scenes" scientific paper written by F. Drago, K. Myszkowski, T. Annen and N. Chiba. The original purpose of the algorithm is to provide a high-quality tone mapping technique to display high contrast images on devices with limited dynamic range of luminance values.

To achieve this goal the operator calculates luminance of every pixel and maximum luminance of the whole image, then divides them by average luminance and finally multiplies the result by user defined exposure factor that is called bias. The bias power function is performed to the whole image splitted into a set of 3×3px tiles (for efficiency sake).

Drago extends the logarithmic response curves in order to handle a wider dynamic range. A logarithmic compression is applied to the image luminance. The base of the logarithm is varied between 2 and 10, based on the brightness of regions within the image. This results in a preservation of contrast in darker regions and a higher compression for bright regions

The main components in Drago’s tone mapping are:

1. Logarithmic compression of luminance values
2. Imitating the human response to light

The main advantages with the system are:

1. Good preservation of details and contrast
2. Gamma correction procedure to improve contrast in dark areas

The main disadvantage is it is slow.

# 3.2 PROPOSED SYSTEM

**3.2.1 SYSTEM ARCHITECTURE**

The overall system architecture is represented in the form of flow chart given below

DATASET OF LDR IMAGES

Images with different exposures

LCG-HDR TECHNIQUE

HDR radiance map, scaling factor

IMPROVED BITONIC FILTER TONE MAPPING

TONE MAPPED HDR IMAGE

**Figure 3.4:** flow chart to generate Tone mapped HDR image

The above pipeline describes how our proposed method works. First the dataset of images which are taken with different exposures are sent to LCG-HDR technique where it produces an HDR radiance map by combining the given input images and returns a scaling factor which is used in tone mapping to proportionate the components of HDR image generated. Then the HDR radiance map is passed through Improved Bitonic Tone Mapping Technique and it generates a HDR image that can be displayed on LDR displays.

The LCG-HDR technique architecture is represented in the following block diagram

LUMINANCE WEIGHT FUNCTION Wy ()

CHROMINANCE WEIGHT FUNCTION Wuv ()

loop for each image

folder

CALCULATE HDR RADIANCE MAP, SCALING FACTOR

IMAGE CONVERSION TO YUV SPACE

APPLY OPERATIONS

READ LDR IMAGES

EXTRACT EXPOSURE VALUES

HDR RADIANCE MAP,  
SCALING FACTOR

COLLECT LUMINANCE SAMPLES

SOLVE CAMERA RESPONSIVE FUNCTION g

CAMERA WEIGHT FUNCTION WCAM

**Figure 3.5:** Block diagram ofLCG HDR

The Figure 3.3 describes the rough sketch of what’s happening inside LCGHDR technique. LCGHDR accepts folder that contains images as input and it reads every image file in the folder. For LCGHDR to work it need only one camera responsive function of luminance channel, for this the system requires to extract sample values from images at random pixels same in all the images of only luminance channel and it also requires exposures extracted from these images and aligned then these values are passed to the modified variant of debevec camera responsive solution for system of equations to produce CRF. This CRF and a couple of weighing function defined and YUV values extracted from each image are used to produce HDR radiance map and scaling factor. The scaling factor when applied reduces the Luminance component to the suitable range of Chrominance channels.

The Improved Bitonic filter Tone mapping technique architecture is represented in the following block diagram.

DECOMPOSE Y-COMPONENT INTO LB & LD  
USING IMPROVED BITONIC FILTER

MERGE  
OPERATION

COMPRESS LB USING GLOBAL TONE MAPPING

HDR IMAGE  
104 : 10-4

AMPLIFY LD

TONEMAPPED HDR IMAGE

**Figure 3.6:** Block diagram ofIBTM process

The Figure 3.3 show the Block diagram of IBTM process which happens after production of HDR radiance map by LCGHDR techniques. This HDR radiance map has values ranging from 10-4 to 104 which can not be processed by general LDR display who’s input values range from 0 to 255, so this HDR radiance map have to be tone mapped. IBTM first decompose the luminance component Y into Lb & Ld components based on the formulae in the below sections. Then Lbis compressed and Ld is amplified and they both are merged to form a Tone mapped image.

**3.2.2 ALGORITHMIC ILLUSTRATION**

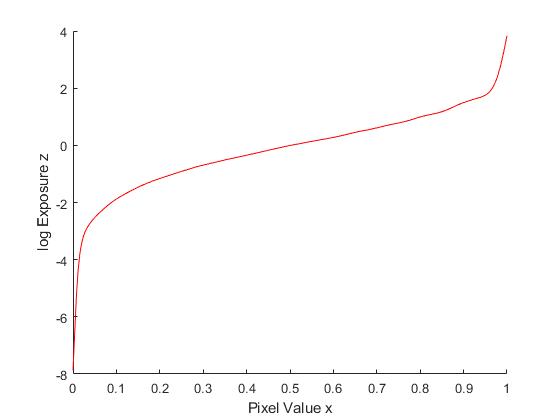
We Start the algorithmby Capturing the image with different either raw or compressed. These images are passed through a series of pipelines as explained in system architecture. The algorithm can be divided into two parts image merging or fusion using LCGHDR method and Tone mapping using IBTM technique. The final output will be an enhanced HDR image that is ready to be displayed.

**3.2.2.1 LUMINANCE-CHROMINANCE COLOR SPACES**

**3.2.2.1 LCGHDR**

The images from the database used as an input and the features such as luminance, chrominance and gradient were extracted from those images. Tone mapping developed based on the features and provide the image with the proper luminance. The higher luminance image and lower luminance images used as input. This technique requires only two input images of higher and lower exposure image and result obtained with exposure with more clarity of the image. The illustration of the dataset and algorithm of this method is given below.

**3.2.2.1.1 CAMERA RESPONSE FUNCTION**

The image a camera acquires consists of a collection of measurements we refer to as intensity values. At a single point in the image, an intensity value is related to the scene radiance by a nonlinear function called the camera response function. A typical camera response has a variation across the image which is linear in scene radiance. Once the response is found, this variation may be calibrated separately and removed. By determining the response, or rather the inverse of the response, we can obtain scene radiance from image intensity. The proposed method requires only to calculate camera responsive function for only luminance channel where as other methods require to calculate for all 3 channels in RGB space. We used a modified version of debevec’s camera responsive solution for our method. The following plot is camera responsive function calculate for luminance channel.

**Figure 3.7** Camera responsive curve for luminance channel

# 3.2.2.1.2 HDR IMAGE COMPOSITION

Let’s take a set of images **ζ**i = [ζiY , ζiU , ζiV ] , i = 1, . . , N in the luminance chrominance space with different exposures assuming **ζ**(x) ∈ [0, 1] × [−0.5, 0.5]2 ,where x=[x1,x2] is a coordinate pixel. The main aim is to obtain single HDR image. In our model the luminance chrominance channels are treated separately for chrominance channel we use saturation weighting is applied. Whereas for luminance channel a pre-calibrated responsive function is used.

# 3.2.2.1.3 LUMINANCE COMPONENT COMPOSITION

In order to obtain camera responsive function for luminance channel a set of images of a scene with different exposures. A good number of pixels which are suitable are chosen, especially pixels having under and over exposed are chosen. From the pixels which are chosen the camera responsive function is fitted in a similar manner of HDR image composition. The camera responsive function is calculated only once and used for linearization of input values in all HDR compositions of same device. In order to obtain HDR luminance component the pixelwise weighted averages of input luminance’s are obtained. Gaussian function (WY) is used as weighing function with standard deviation of 0.2 and mean of 0.5 thus making sure of a small impact of under are over-exposed pixels. The logarithmic HDR luminance component is obtained as

|  |  |  |
| --- | --- | --- |
|  |  |  |

(1)

From the eq [1] value of HDR luminance is measured in logarithmic scale and camera responsive function is g. Once the natural exponential is applied the resulting values are spanned normally and positive .

# 3.2.2.1.4 CHROMINANCE COMPONENT COMPOSITION

Color saturation is taken to weight the component where are camera response function is not required to calculate the chrominance component composition. The more the color saturation in the image, the more pixel contains more chromatic information. The chrominance feature is chosen because its values is saturated when pixel is over or under exposed. In detail, where In this method, we also recognised that is a good choice. Same weight is used for both chromatic components to guarantee the color perseveration and compose any chromatic component as

(2)

Saturation of is denoted by .The range of is again in

# 3.2.2.1.5 GRADIENT VALUE OF IMAGE

Weighted map of gradient information is created for static and dynamic scenes and at the same time to canny detection. The gradient information is extracted by first derivatives of 2-D gaussian filter in the x and y directions are given by 15 and 15  which are partial derivatives of images along x and y axis. Two pixels in the experiment are set by standard deviation and gradient magnitude reflects maximum change in the pixel values. The two components are calculated in the following equation

(3)

The use luminance, chrominance and gradient for tone mapping provide the image with proper illumination.

# 3.2.2.1.6 FUSION METHOD

T which is a luminance range reduction operator will define its output as . In terms of chromatic channel, it is the best approach. Rendering of very dark or very bright and saturated colors are not allowed by RGB gamut. This is present in real images captured in HDR images. Hence, there is need for chromatic feature and hue intact in order to fit into RGB, s gamut. The scaling factor is introduced for two chrominances will reduce down the saturation and there will be no change in hue. It uses RGB values which are embedded in color space transformation and describe as follows

The luminance, chrominance and gradient to RGB transformation matrix be and define the chromatic complement image and achromatic(gray) as

by )15

Truly grey image is  ,since in RGB luminance, chrominance and gradient transforms.

Look for a map such that

(4)

This is defined by

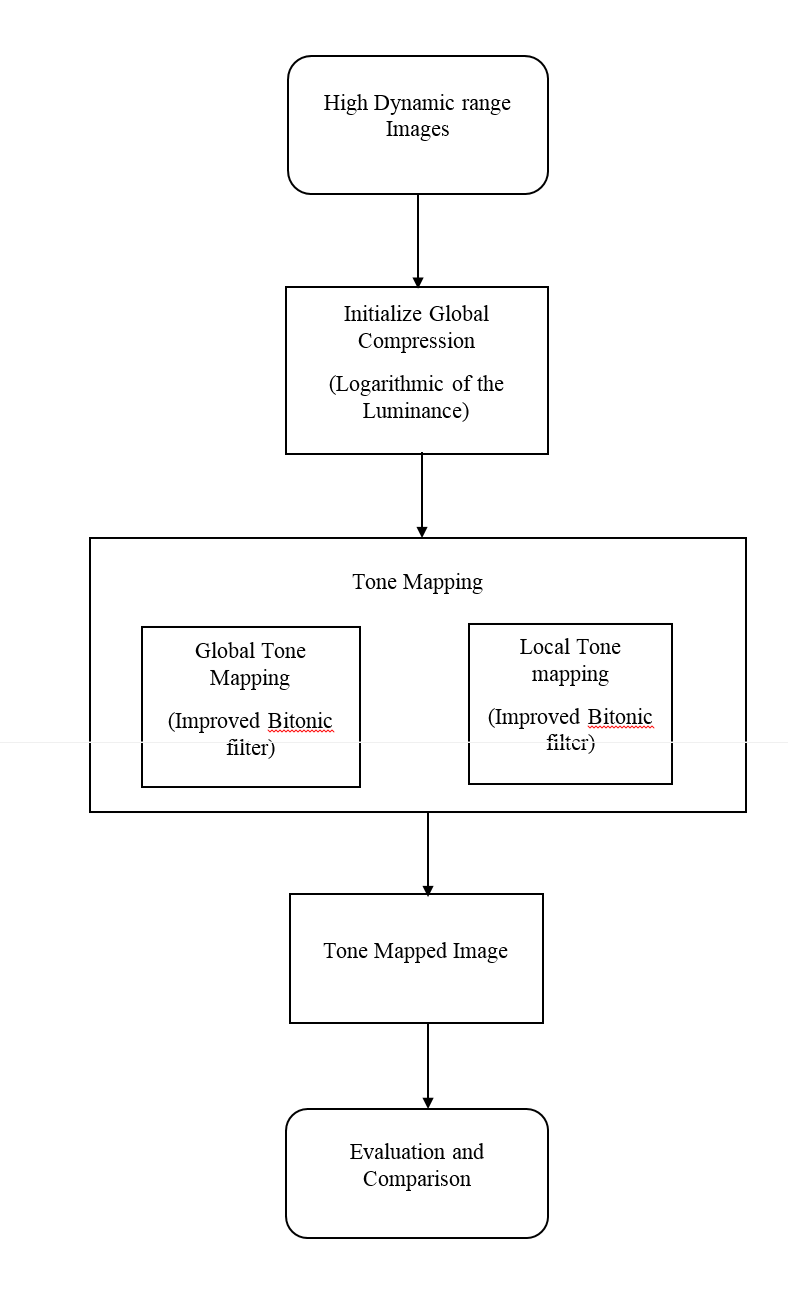
(5)

In the above formula are analogously. The hue of is not influenced by ,on the other hand the saturation is measured proportionally to it. Colours which consist in same hue as those of HDR image are from low dynamic range image , as it needed to fit with RGB gamut. The LCG technique image can be stored directly in an arbitrary method for display which is changed into RGB by using matrix B.

**Algorithm 1.1:** HDR image generation using LCGHDR technique

**Input:** image [1…n] with different exposures  
**output:** HDR image

**for** **each** image Ci  :   
  **if** Ci is in RGB color space**then**  
 convert Ci into YUV space   
 **end if**  
 extract Yi,Cri,Cbi values from Ci  
 calculate Y,Cr,Cb component of final HDR image as per formulae using extracted values  
 calculate scaling factor from extracted values   
**end for each**  
apply exponentiation on Y component of HDR image  
divide Cr and Cb components of HDR image with scaling factor  
**return** HDR image

**3.1.2.2 TONE MAPPING**

**Figure 3.8:** Flow chart of Tone Mapping

The following figure is flow chart of TMO with respect to bitonic filter which is improved. The main goal of tone mapping(reproduction) is to compress the dynamic range of a scene to a range that will be easily displayed on physical devices only when the luminance range of image is broader than physical devices. All tone mapping techniques can be categorized to global and local tone mapping. Global operators apply same technique to every one pixel of a image whereas coming to local operators adapt their scales to different areas of the image. To compress the range of luminance to the range which can be displayed it is initially mapped using global tone mapping function. To increase the quality of the image, “dodging and burning” technique is introduced. Which allows different exposures to each part of the image. In automated process low contrast regions are founded by center surrounded function. Then a tone mapping function is locally applied. While preserving overall characteristics of an image the automatic dodging and burning method enhance contrast and details of image.

Given an HDR image, the luminance component of the HDR image in Equation (1) is decomposed as

Where and are the base layer and the detail layer. The dynamic range of detail layer is small when compared to base layer. Using global tone mapping algorithm the overall tone mapping algorithm is reduced to scale the base layer as. The detail layeris preserved or even amplified as to enhance local contrasts. The compressed luminance valueis the product of and. The HDR image and are finally adopted to generate the output LDR image.

The log function approximates the transformation performed by the retina of the HVS and from this decomposition is performed in the log domain in equation

Where and are and, respectively. Three major function of the proposed technique are: the decomposition of into and, the compression of through global tone mapping and the amplification of.Improved Bitonic filter is used to decomposethe HDR images.

# 3.2.2.3 IMPROVED BITONIC FILTER

In order to eliminate impulsive noise median filter is commonly used whilst preserving edges well, while there is no more than one edge with in the range. A rank filter is a generalization of the median where any centile can form the output. Those filters are considered as monotonic because they store signals which are monotonically increasing or decreasing therefore it leads to noise reduction. The shape of window is used form the set of ranked data for two-dimensional (2D) data. For 2D image disk here we use circular disk to ensure isotropic behavior. If B is a Bitonic filter image and be variance of in the window, . Local variances defines edge-aware weighting of 3×3 windows of all pixels as follows

(6)

Where is defined as small constant and the value of it is given as When dynamic range of input image is given as L. In order to compute all pixels in the image are used. If value of is smaller than 1 then the value of is larger than 1. IT is crystal clear that larger weights are assigned to pixels at edge using ). In order to prevent blocking of artifacts from appearing in final image, the value of is smoothed by a Gaussian filter. The proposed weighting matches one features of human visual system, i.e., pixels at sharp edges are usually more important than those in flat areas.

Where is a rank filter, filter window is w and window length is and c is choosen centile. Robust closing and opening operations half of the required purpose, they only preserve local minima or maxima .In addition to it , that the opening and closing operations do not store mean values of signals in the case of a noisy signal. Both drawbacks can overcome by same means. The most appropriate output for each part of the pixel is clear by comparing each of the original pixels with opening and closed operations. Hence we can use such a comparison to weight a combination of opening and closing operations. , so instead the differences between the original signal and each of the rank-filtered pixel are smoothed with a linear filter. A Gaussian filter (i.e. a linear moving-window filter with Gaussian weights) is used for this purpose, since it is known to have good noise reduction properties. The filter length is determined experimentally to match the noise reduction from the rank filters, so that the standard deviation *σ* = 0*:*33*lwherel* is the window length in 1D, or the diameter of the structuring element in 2D. This smoothed error can be seen in the middle column of

The Gaussian linear filter as, this is used to weight the results of the opening and closing operations as follows:

(7)

are smoothed opening and closing errors and is output of improved bitonic filter.

**4. EXPERIMENTAL ANALYSIS AND RESULTS**

**4.1 SYSTEM REQUIREMENTS**

This section will document the specific functional and non-functional software and hardware requirements of the LCGHDR technique.

**4.1.1 SOFTWARE REQUIREMENTS**

**4.1.1.1 MATLAB**

**MATLAB** is a general-purpose programming language. When it is used to process images one generally writes function files, or script files to perform the operations. These files form a formal record of the processing used and ensures that the final results can be tested and replicated by others should the need arise. In our project we are using Matlab to build our code and to perform the experimental operations, ultimately showing our intended output in Matlab.

You need have a MATLAB of version 2010 or above to process the LCGHDR.Matlab allows its users to accurately solve problems, produce graphics easily and produce code efficiently. MATLAB’s disadvantages are it is an interpreted language, it can be slow, and poor programming practices can make it unacceptably slow.

**4.1.1.2 HDR TOOL BOX**

The HDR Toolbox provides functions for processing HDR images and videos for different tasks such as tone mapping, inverse tone mapping, reverse tone mapping, expansion, HDR compression, image-based lighting, handling HDR videos, color transforms, etc. The HDR Toolbox is part of the book "Advanced High Dynamic Range Imaging", where you can find extra documentation about the code and more details on HDR imaging. HDR tool box resided in the MATLAB itself.

HOW TO INSTALL:  
1) Unzip the file HDRToolbox.zip in a FOLDER on your PC/MAC   
2) Run Matlab   
3) Set the FOLDER as the current directory   
4) Write the command: “install HDRToolbox” in the Command Window, and wait for the installation process to end.

**4.1.1.3 DCRAW**

DC raw tool is used to compress the raw image into RGB format.Dcraw is an open source computer program which is able to read numerous raw image format files, typically produced by mid-range and high-end digital cameras. Dcraw converts these images into the standard [TIFF](https://en.wikipedia.org/wiki/Tagged_Image_File_Format) and [PPM](https://en.wikipedia.org/wiki/Netpbm_format) image formats. This conversion is sometimes referred to as developing a raw image since it renders raw image sensor data (a "digital negative") into a viewable form.

A number of other image processing programs use Dcraw internally to enable them to read raw files.

**4.1.1.4 EXIF TOOL:**

Exif tool is used to extract data by taking tag id as an input. Every image like thumbnails, compressed and uncompressed images have their own tag ids taking this tag id as input Exif tool generates the data about the pic.Exif Tool is a free and open-source software program for reading, writing, and manipulating image, audio, video, and PDF metadata. It is platform independent, available as both a Perl library and command-line application.

**Steps to run Exif tool from the command line:**

1. Click the Windows "Start" menu and run the "cmd" application
2. Type "exiftool" followed by a SPACE on the cmd window.
3. Drag and drop files and folders on the cmd window.
4. Press RETURN to view the metadata from the files you dropped.

**4.1.2 HARDWARE REQUIREMENTS**

**Recommended System Requirements:**

Processors : Intel® CoreTM i5, 2.60 GHz  
 RAM : 8 GB DRAM  
 Disk space : 20 to 30 GB  
 Operating systems : Windows® 10, mac OS\*, and Linux\*

**Minimum System Requirements:**

Processors : Intel® Core2duo or Pentium 4, 2.60 GHz  
 RAM : 4 GB DRAM  
 Disk space : 15 to 18 GB  
 Operating systems : Windows® 10, mac OS\*, and Linux\*

**4.2 SAMPLE CODE**

**Source code:**

**File:** lcghdr.m

1. % Generates a hdr radiance map from a set of pictures
2. % parameters:
3. % filenames: a list of filenames containing the differently exposed pictures used to make a hdr from
4. % gLuminance: camera response function **for** the luminance channel
5. % resolution: image dimensions
6. % weightY : weighing function **for** wY **in** formulae refer paper
7. % dt: log exposure values of images
8. % output:
9. % hdr: hdr image [10^-4,10^4] x 1,[-0.5,0.5] x 2
10. % scaleFactor: it defines the adjustment btw luminance **and** chrominance channels
12. function [ hdr, scaleFactor] = lcgHdr (filenames, resolution, gLuminance, weightY, dt)
14. numExposures = length(filenames);
15. rgbToYcbcrMat = [0.30,-0.17,0.50; 0.59, -0.33, -0.42; 0.11, 0.50, -0.08];
17. % pre-allocate resulting hdr image
19. hdr = zeros(resolution);
21. %pre-allocate equation components
22. luminanceDenominator = zeros(resolution(1),resolution(2));
23. luminanceNumerator = zeros(resolution(1),resolution(2));
24. CvNumerator = zeros(resolution(1),resolution(2));
25. CuCvDenominator = zeros(resolution(1),resolution(2));
26. CuNumerator = zeros(resolution(1),resolution(2));
27. scaleFactorDenominator=zeros(resolution(1),resolution(2));

30. **for** i=1:numExposures
32. fprintf('Adding picture %i of %i \n', i, numExposures);
34. %read picture converting it to tiff **class**
35. imagePath= convertStringsToChars(filenames(i).folder + '\' + filenames(i).name);
37. %---------------- **for** raw **and** uncompressed reading---------------%
38. %         tiffImage = Tiff(imagePath,'r');
39. %         offsets = getTag(tiffImage,'SubIFD');
40. %         setSubDirectory(tiffImage,offsets(3));
41. %         %read Y cb cr components
42. %         [image(:,:,1),image(:,:,2),image(:,:,3)] = read(tiffImage);
43. %         close(tiffImage);
44. %-----------------**for** compressed **and** sRGB files-------------------%
45. rgbDouble=im2double(imread(imagePath));
46. r=rgbDouble(:,:,2);
47. %-----------------conversion rgb to ycbcr---------%
49. ycbcrDouble = reshape(rgbDouble,resolution(1)\*resolution(2) , 3) \* rgbToYcbcrMat;
50. ycbcrDouble = reshape(ycbcrDouble, resolution(1),resolution(2), 3);
51. y=im2uint8(ycbcrDouble(:,:,1));
52. %-------/Luminance composition calculation/----------
54. WyOfYi = weightY(y + 1);
55. luminanceDenominator = luminanceDenominator + WyOfYi ;
56. luminanceNumerator = luminanceNumerator+ ( WyOfYi .\* (gLuminance(y + 1)-dt(1,i)));
58. %----------/cb Chrominance U composition/------------
60. WySi = sqrt(Cb.\*Cb + Cr.\*Cr);
61. WySi=WySi.^ (1.5);
62. CuNumerator = CuNumerator + (WySi .^ ycbcrDouble(:,:,2));
63. CuCvDenominator = CuCvDenominator + WySi;
65. %--------/ cr Chrominace V composition /--------------
67. CvNumerator = CvNumerator + (WySi .^ ycbcrDouble(:,:,3));
68. %scaleFactor
69. scaleFactorDenominator= scaleFactorDenominator + (WySi .\* Yi );
71. **end**
73. hdr(:,:,1) = luminanceNumerator ./ luminanceDenominator;
74. hdr(:,:,2) = (CuNumerator ./ CuCvDenominator);
75. hdr(:,:,3) =(CvNumerator ./ CuCvDenominator);
77. %apply exponential to hdr
79. hdr(:,:,1)= exp(hdr(:,:,1));
80. scaleFactor = (hdr(:,:,1).\* CuCvDenominator) ./ scaleFactorDenominator ;
82. **end**

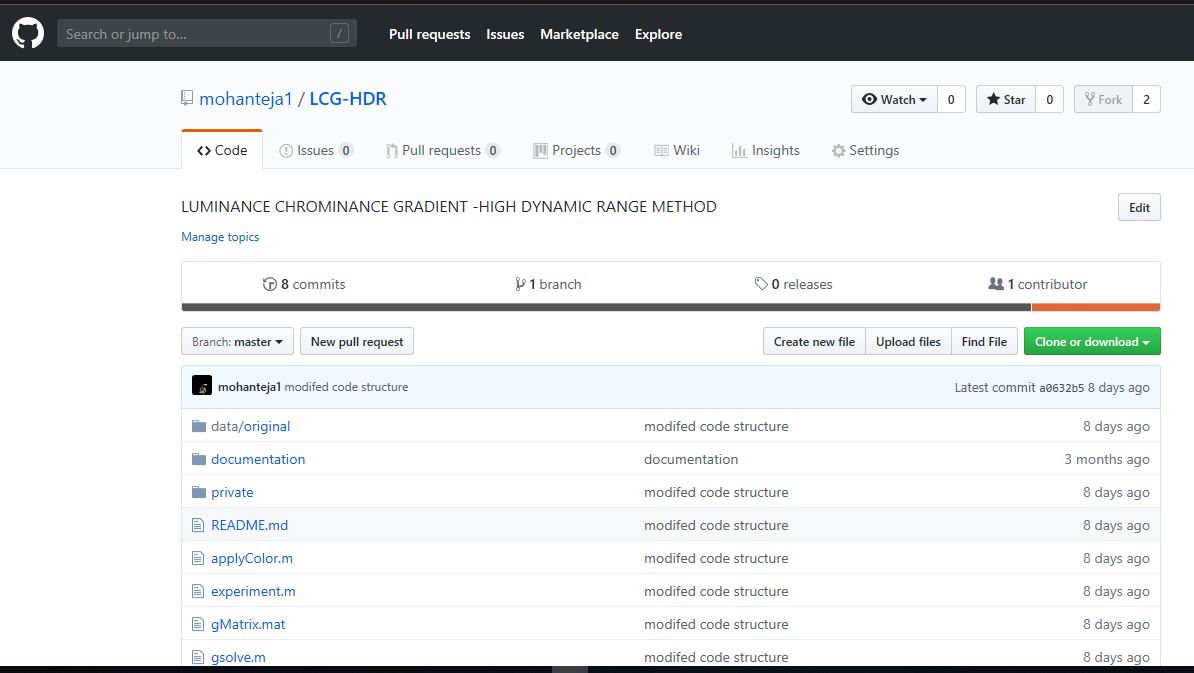
**File:** lcgWeights.m

1. function w = lcgWeight(pixelValue, alpha, beta,CamOrY)
2. % calculates weight given  w = p^a \* (1-p)^b
3. % **for** Wcam function CamOrY = 1
4. % **for** Wy function CamOrY =0
5. **if** CamOrY==1
6. w = 17.6 \*( pixelValue ^ alpha \* ((1-pixelValue)^beta));
7. **else**
8. w= 63.5 \*( pixelValue ^ alpha \* ((1-pixelValue)^beta));
9. **end**

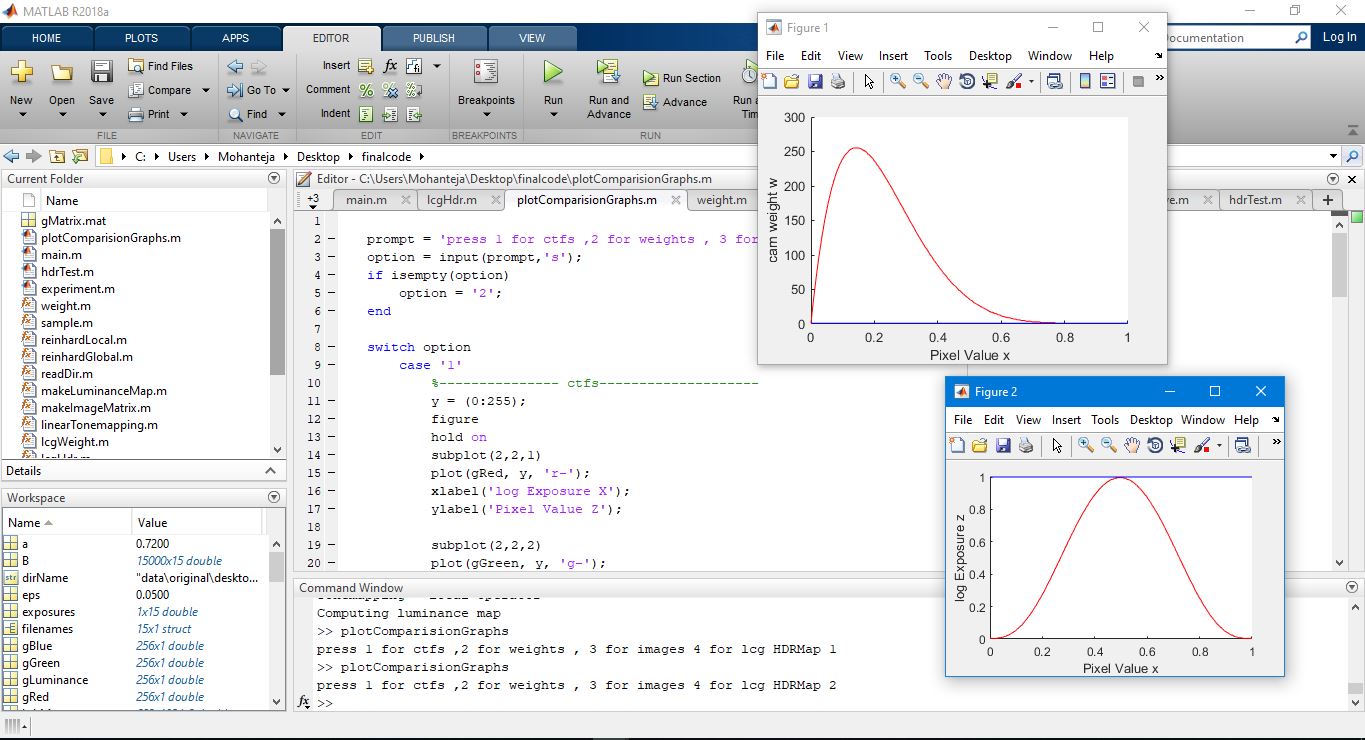
**File:** gSolve.m

1. function [g,lE]=gsolve(Z,B,l,w)
2. n = 256;
3. A = zeros(size(Z,1)\*size(Z,2)+n+1,n+size(Z,1));
4. b = zeros(size(A,1),1);
6. %% Include the data-fitting equations
7. k = 1;
8. **for** i=1:size(Z,1)
9. **for** j=1:size(Z,2)
10. wij = w(Z(i,j)+1);
11. A(k,Z(i,j)+1) = wij;
12. A(k,n+i) = -wij;
13. b(k,1) = wij \* B(i,j);
14. k=k+1;
15. **end**
16. **end**
18. %% Fix the curve by setting its middle value to 0
19. A(k,129) = 1;
20. k=k+1;
22. %% Include the smoothness equations
23. **for** i=1:n-2
24. A(k,i)=l\*w(i+1); A(k,i+1)=-2\*l\*w(i+1); A(k,i+2)=l\*w(i+1);
25. k=k+1;
26. **end**
28. %% Solve the system using SVD
29. x = A\b;
30. g = x(1:n);
31. lE = x(n+1:size(x,1));

**4.3 SCREENSHOTS**

****

**Figure 4.1:** Uploaded screenshot from GitHub

****

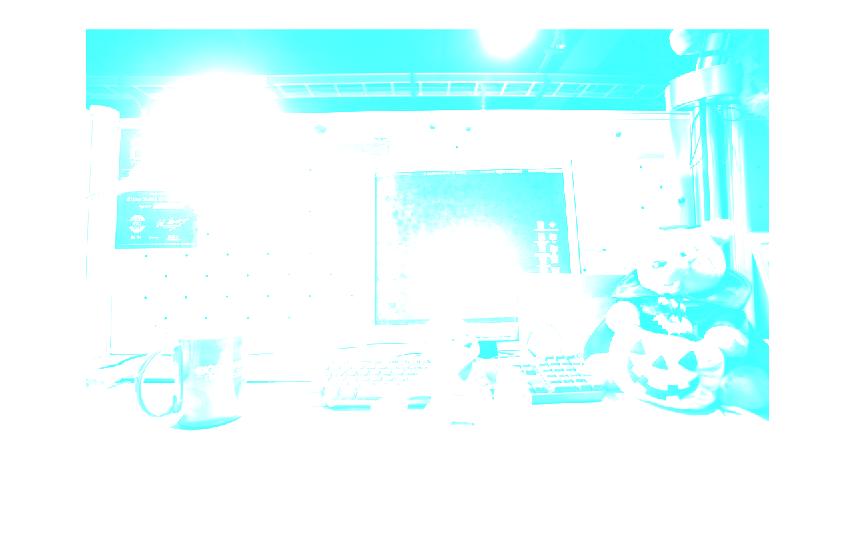
**Figure 4.2:** matlab program showing results formed (weight function)

****

**Figure 4.3:** matlab program showing results formed (camera response function)

**4.4 EXPERIMENTAL ANALYSIS**

Using MATLAB, the experiment is conducted where the system specifications must be 3.0 GHz processor and 4 GB ram. In the same environment efficiency analysis is performed for all methods. From the available HDR photographic survey website the input images are collected. Output image produced is a perfectly illuminated image which was taken for evaluation. In terms of signal to noise ratio (SNR) our method performed best. Each set of images have the fifteen similar images captured with different exposures. The HDR techniques used for the 10 different set of images and obtain the image with proper illumination

There are 8 images used as input for evaluation in our analysis for picture quality. As illuminance, chrominance and gradient values extracted from images. Table 4.1 explains time taken by different methods on different scenes with different images count.

**Figure 4.4**: HDR radiance map of YUV image formed by LCGHDR showed in RGB space

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |

**Figure 4.5**: series of images with different exposure levels taken as input.

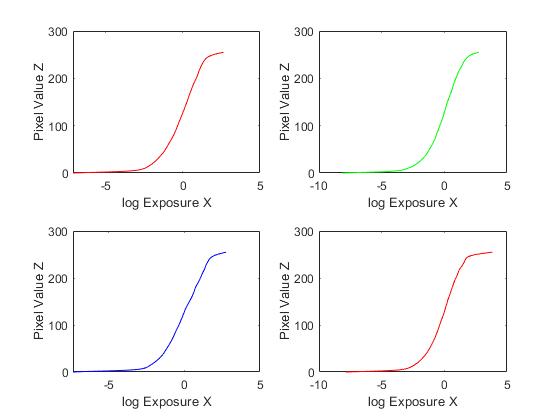
|  |
| --- |
| C:\Users\Dell\Downloads\lcghdrIBTM.png |
| (a) |
| C:\Users\Dell\Downloads\durand.jpg |
| (b) |
| C:\Users\Dell\Downloads\reinhard.jpg |
| (c) |

**Figure 4.6**:

(a) output of our proposed technique (LCGHDR with IBTM) (b) output using Durand technique (c) output using Reinhard technique**.**

Table 1. 4.1 Comparison of BAR, AEE and LCGHDR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Number of scenes | Methods | Number of shots | % Lost |  |
| 1 | Durand | 3 | 1.61 | 0.301 |
| Reinhard | 2 | 1.61 | 0.025 |
| LCGHDR with IBTM | 2 | 1.61 | 0.014 |
| 2 | Durand | 3 | 0 | 0.301 |
| Reinhard | 2 | 0 | 0.101 |
| LCGHDR with IBTM | 2 | 0 | 0.85 |
| 3 | Durand | 3 | 0 | 0.301 |
| Reinhard | 2 | 0 | 0.04 |
| LCGHDR with IBTM | 2 | 0 | 0.021 |
| 4 | Durand | 3 | 0 | 0.301 |
| Reinhard | 2 | 0 | 0.05 |
| LCGHDR with IBTM | 2 | 0 | 0.032 |
| 5 | Durand | 3 | 0 | 0.301 |
| Reinhard | 2 | 0 | 0.034 |
| LCGHDR with IBTM | 2 | 0 | 0.015 |
| 6 | Durand | 3 | 0 | 0.301 |
| Reinhard | 2 | 0 | 0.025 |
| LCGHDR with IBTM | 2 | 0 | 0.012 |

****

**Figure 4.7:** Comparison of camera response functions of R, G, B, Y components

**5. CONCLUSION AND FUTURE WORK**

**5.1 CONCLUSION**

The combination of LCGHDR pipeline with Improved Bitonic Filter Tone Mapping technique generates pleasant noise free high dynamic range images. Since LCGHDR technique to produce HDR luminance map takes less time than the available methods like Reinhard, Durand, Drago etc. The IBTM technique produces noise free tone mapped HDR images and our method uses both of these techniques making it surpass the existing methods in terms of time complexity and noise ratios and quality of image. But this method takes slightly more memory than existing methods. Since we have implemented this technique in MATLAB, it is possible to reduce the memory intake by the method if it is implemented in native programming language like cpp. Objective comparison of computational efﬁciency is difﬁcult due to, among others, differences in implementation and execution environments and optimization of the different methods. However, in terms of complexity, the presented approach is not less efﬁcient than any RGB-based one.

**5.2 FUTURE WORK**

In near future we will tweak our model to enhance the shortcomings of the earlier work and also, we will implement a real time working hardware model that will capture the images in raw and uses our method to develop HDR images within fraction of seconds. We will also define a python package for our method that can be imported by others which can be helpful for research purpose.

**6. REFERENCES**

1. Song, Y., Jiang, G., Yu, M., Peng, Z. and Chen, F., 2018. Quality assessment method based on exposure condition analysis for tone-mapped high-dynamic-range images. Signal Processing, 146, pp.33-40.
2. Fan, W., Valenzise, G., Banterle, F. and Dufaux, F., 2018. Fine-Grained Detection of Inverse Tone Mapping in HDR Images. Signal Processing.
3. Ambalathankandy, P., Horé, A. and Yadid-Pecht, O., 2016. An FPGA implementation of a tone mapping algorithm with a halo-reducing filter. Journal of Real-Time Image Processing, pp.1-17.
4. Schulz, S., Grimm, M. and Grigat, R.R., 2007. Using brightness histogram to perform optimum auto exposure. WSEAS Transactions on Systems and Control, 2(2), p.93.
5. Grosch, T., 2006. Fast and robust high dynamic range image generation with camera and object movement. Vision, Modeling and Visualization, RWTH Aachen.
6. Paul E. Debevec and Malik “Recovering high dynamic range radiance maps from photographs” SIGGRAPH '97 Proceedings of the 24th annual conference on Computer graphics and interactive techniques Pages 1997 369-378
7. Graham treece bitonic filter Published in Journal IEEE Transactions on Image Processing archive Volume 25 Issue 11, November 2016 Pages 5199-5211 IEEE Press Piscataway, NJ, USA.

**ONLINE RESOURCES:**

1. MITs new HDR video algorithm: <https://gadgets.ndtv.com/cameras/news/mit-develops-real-time-hdr-camera-algorithm-to-prevent-overexposure-731675>
2. Teo de Campos page: <http://www.robots.ox.ac.uk/~teo/>
3. HDR image database: <http://hdrplusdata.org/>
4. Colin Doutre’s page: <http://www.ece.ubc.ca/~colind/>
5. Google’s HDR + implementation: http://timothybrooks.com/tech/hdr-plus/
6. Introduction to Visual Computing: https://www.cs.toronto.edu/~mangas/teaching/320/calendar.html
7. Cotter’s HDR tools: https://ttic.uchicago.edu/~cotter/projects/hdr\_tools/
8. Paulbourke’s page: http://paulbourke.net/
9. Plataniotis paper for tone mapping operator: https://www.comm.utoronto.ca/~kostas/
10. Digital image processing class: http://www.cs.umsl.edu/~sanjiv/classes/cs5420/
11. Evaluation of tone mapping operators :http://cadik.posvete.cz/tmo/
12. Durand implementation: http://vision.gel.ulaval.ca/~jflalonde/cours/4105/h14/tps/results/tp5/minghou/index.html
13. G treece bitonic filter: https://www.repository.cam.ac.uk/bitstream/handle/1810/252987/treece\_tr700.pdf?sequence=1&isAllowed=y
14. Durand bilateral filter for hdr: <https://people.csail.mit.edu/fredo/PUBLI/Siggraph2002/DurandBilateral.pdf>
15. Web hdr: <http://www.jaloxa.eu/webhdr>
16. Dragos paper: www.resources.mpi-inf.mpg.de/tmo/logmap/
17. Erik Reinhard website: [www.erikreinhard.com/hdr.html](http://www.erikreinhard.com/hdr.html)
18. Reinhard hdr implementation: [www.cybertron.cg.tu-berlin.de/eitz/hdr/index.html](http://www.cybertron.cg.tu-berlin.de/eitz/hdr/index.html)
19. luminance-chrominance approach: <http://www.cs.tut.fi/~hdr/#ref_problems>
20. camera:algorithms:<https://www.eecs.tuberlin.de/fileadmin/fg144/Courses/10WS/pdci/talks/camera_algorithms.pdf>
21. pfstools for hdr: www.pfstools.sourceforge.net/pfstmo.html
22. Computational photography Udacity: https://in.udacity.com/course/computational-photography--ud955

**GITHUB REFERENCE CODES:**

1. vfx-hdr:(Reinhard) <https://github.com/hsiaoyi0504/vfx-HDR/tree/master/submit/src>
2. "A Bio-Inspired Multi-Exposure Fusion Framework for Low-light Image Enhancement:   
    link: https://github.com/baidut/BIMEF
3. C++ code for finding camera response function: <https://github.com/cbraley/hdr>
4. Our project code is available at: <https://github.com/mohanteja1/LCG-HDR>

**APPENDIX**

HDR High Dynamic Range

LDR Low Dynamic Range

MATLAB Matrix Laboratory

RGB values Red Green Blue values

ISO International Organization of Standardization

LCD Liquid Crystal Diode

CRT Cathode Ray Tube

TMO Tone Mapping Optimization

IBTM Improved Bitonic filter Tone Mapping

LCGHDR Luminance Chrominance Gradient High Dynamic Range

DCRAW images Decoding RAW images

EXIF tool Exchangeable Image File format