



## ESTUDO REFERENTE À VISÃO POR GRANDE ALCANCE DINÂMICO, UTILIZANDO O CONCEITO DE HDR

Felipe Sens Bonetto<sup>1</sup>, Johann Schmitdinger Vieira<sup>2</sup>, Vitor Pereira Mattos<sup>3</sup>, Fábio Cabral Pacheco<sup>4</sup>, Jony Laureano Silveira<sup>5</sup>, Leandro Schwarz<sup>6</sup>, Hugo Marcondes<sup>7</sup>, Fernando Pedro Henriques de Miranda<sup>8</sup>

**Resumo:** Cada vez mais imersa no mercado, a captura de imagens em High Dynamic Range (HDR) torna-se uma técnica que atrai a atenção para pesquisas em diversas áreas de aplicação, como por exemplo, em controles de processos industriais e operacionais. O HDR é um método de captura de imagem digital que registra com qualidade um maior alcance dinâmico de luminosidade do que a captura de imagens em processos comuns, mantendo numa mesma imagem um baixo contraste ao longo dos diversos pontos luminosos. Este artigo apresenta o desenvolvimento de um algoritmo para criação de imagens em HDR, escrito em linguagem de programação C com uso da biblioteca de processamento de imagem do software OpenCV, para o projeto desenvolvido pelo Grupo de Pesquisa e Desenvolvimento em Sistemas Embarcados (GPDSE) do Instituto Federal de Ciência e Tecnologia de Santa Catarina. Este trabalho é sustentado por um referencial teórico que abrange desde conceitos ópticos relevantes até o conceito de HDR. Também serão abordadas as técnicas de tratamento de imagens, que geraram resultados satisfatórios.

**Palavras-chave:** HDR. Processamento de Imagem. FPGA. ARM.

**Abstract:** *Increasingly immersed in the market, the capture of High Dynamic Range images quickly became the center of much attention for research in several fields of application, such as in the control over industrial and operational activities. HDR is a method of digital image capture that is capable of registering a scene with a higher luminosity dynamic range than standard procedures, maintaining in a single image a low contrast between areas with different luminosity concentrations. This article presents the development of an algorithm for the creation of HDR images, written in C programming language using the OpenCV image-processing library software, for the research project by the Research Group of Embedded Electronic Systems (GPDSE) of the Federal Institute of Science and Technology of Santa Catarina. This research is supported by a theoretical reference that goes from the optical concepts relevant to HDR to the concept of HDR itself. There will also be an approach to practical HDR imaging capture techniques, generating relevant results for the research.*

**Keywords:** HDR. Image Processing. FPGA. ARM.

<sup>1</sup> Acadêmico do Curso Técnico Integrado em Eletrônica, *campus* Florianópolis, IFSC <felipesensbonetto@gmail.com>.

<sup>2</sup> Acadêmico do Curso Técnico Integrado em Eletrônica, *campus* Florianópolis, IFSC <johanns.vieira@gmail.com>.

<sup>3</sup> Acadêmico do Curso Superior de Tecnologia em Sistemas Eletrônicos, *campus* Florianópolis, IFSC <vitor.pm@aluno.ifsc.edu.br>.

<sup>4</sup> Técnico em Eletrônica do Departamento Acadêmico de Eletrônica (DAELN), *campus* Florianópolis, IFSC <fabio.pacheco@ifsc.edu.br>.

<sup>5</sup> Professor do Departamento Acadêmico de Eletrônica (DAELN), *campus* Florianópolis, IFSC <jony@ifsc.edu.br>.

<sup>6</sup> Professor do Departamento Acadêmico de Eletrônica (DAELN), *campus* Florianópolis, IFSC <schwarz@ifsc.edu.br>.

<sup>7</sup> Professor do Departamento Acadêmico de Eletrônica (DAELN), *campus* Florianópolis, IFSC <hugo.marcondes@ifsc.edu.br>.

<sup>8</sup> Professor do Departamento Acadêmico de Eletrônica (DAELN), *campus* Florianópolis, IFSC <fernando.miranda@ifsc.edu.br>.

### 1. INTRODUÇÃO

O presente documento descreve o procedimento inicial de pesquisa realizado para o projeto intitulado ViGAS (Visão por Grande Alcance dinâmico para Soldadores), do Grupo de

Pesquisa e Desenvolvimento em Sistemas Embarcados (GPDSE) situado no Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina (IFSC).

Abordando o método de captura e processamento de imagem em *High Dynamic Range*

(HDR), esta pesquisa fornece uma base teórica para o entendimento dessa tecnologia por meio de um referencial teórico e relaciona os meios práticos de se utilizar HDR, portanto, é capaz de determinar um rumo do projeto em função do objetivo desejado.

Com base na pesquisa desenvolvida, foi possível a elaboração de um algoritmo computacional capaz de executar o procedimento de criação de uma imagem HDR. O método escolhido foi o de superposição de imagens *Low Dynamic Range* (LDR). Este algoritmo foi implementado utilizando a linguagem de programação C e a biblioteca de processamento de imagens OpenCV. O procedimento é detalhado com exemplificação gráfica, fluxogramas e uso de código fonte. As imagens em HDR obtidas serão comentadas e discutidas.

## 2. REFERENCIAL TEÓRICO

*High Dynamic Range* (grande alcance dinâmico) ou HDR, como visto anteriormente, é um método que promove a captura de imagem digital e realiza o registro com uma excelente qualidade, um maior alcance dinâmico de luminosidade, comparado com as capturas de imagens comuns. Este método mantém numa mesma imagem um baixo contraste ao longo dos diversos pontos luminosos.

### 2.1. Fundamentos ópticos utilizados no HDR

#### 2.1.1. Exposure Value

Para diferentes situações de luminosidade, a câmera se adapta para adquirir a melhor imagem. Ela faz isso ajustando seu *Exposure Value* (valor de exposição), conhecido como EV. O EV depende essencialmente de dois mecanismos da câmera, que serão explicados em seguida, mas de maneira geral cada paisagem exige um EV diferente para uma captura fiel.

O EV é diretamente relacionado à quantidade de luz que chega à câmera. Por exemplo, em um dia de Sol na praia, a areia reflete grande parte da luminosidade a que é submetida. Com bastante luz chegando à câmera, esta se adapta para um alto EV (para esta situação, um EV de aproximadamente 16). Em outro caso, para capturar uma imagem da aurora boreal, é utilizado um EV baixo (EV de -6 à -4), pois ela representa pouca luminosidade num céu escuro pela noite.

O EV é determinado por duas grandezas, sendo estas o tempo de exposição ( $t$ ) do sensor e a razão focal ( $N$ ) da lente, que pode ser determinado pela Equação 1.

$$EV = \log_2 \frac{N^2}{t} \quad [1]$$

#### 2.1.2. Tempo de Exposição

O tempo de exposição luminoso ao qual o sensor é submetido define a qualidade para uma determinada luminosidade. Esse tempo pode ser controlado de forma eletrônica ou pelo uso de um obturador. Um baixo tempo de exposição resulta numa boa qualidade para áreas com alta luminosidade, da mesma maneira que um alto tempo de exposição implica em melhor qualidade em áreas mais escuras. A Figura 1 demonstra uma mesma paisagem fotografada com tempos de exposição baixo (Figura 1(a)), médio (Figura 1(b)) e alto (Figura 1(c)).

Relacionando o tempo de exposição com o EV, um alto tempo de exposição resulta em um menor EV. Independentemente do tempo de exposição utilizado, nota-se que em todas as situações existe um forte contraste entre os planos luminosos das imagens. O céu claro e a baía escura são dois planos com luminosidades bem diferentes dentro de uma mesma imagem, o que para um tempo de exposição fixo resulta no contraste observado.

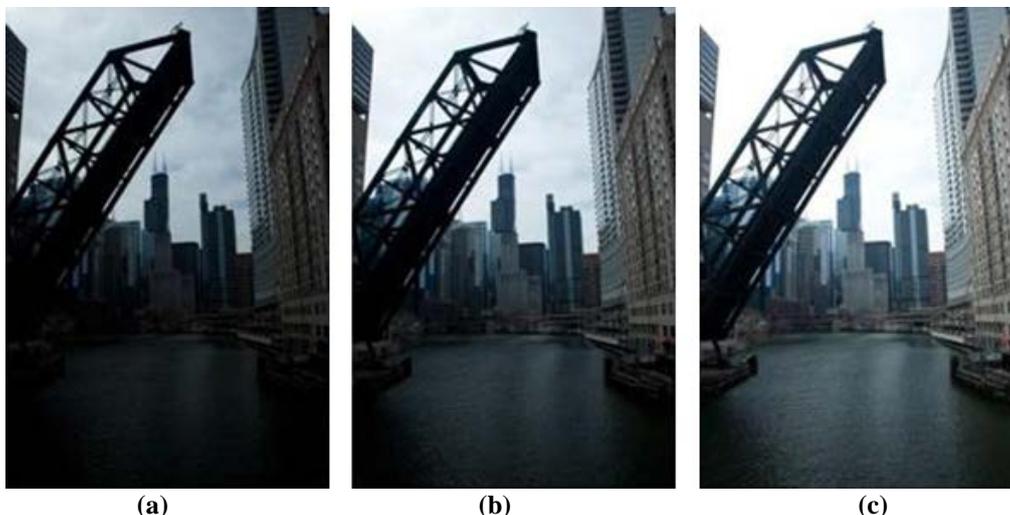


Figura 1 – Paisagem em diferentes tempos de exposição. Imagem de Mark Stagi.

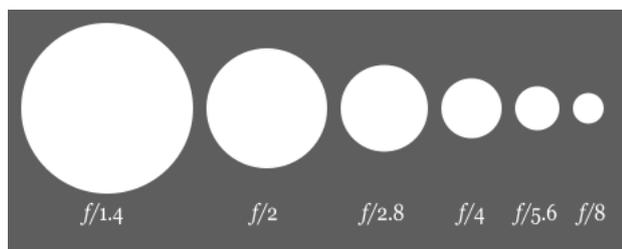
Adicionalmente, o tempo de exposição também determina o *motion blur* (desfoque de movimento) da imagem. Um menor tempo de exposição resulta numa imagem de movimento estático, enquanto alto tempo de exposição desfoca os objetos em maior movimento.

### 2.1.3. Razão focal

O outro fator que influencia na captura de luminosidade, além do tempo de exposição, é o raio focal da lente. Entende-se por raio focal a razão entre a distância de foco e o diâmetro da lente, conforme mostra a Equação 2, que relaciona a distância de foco ( $f$ ) com o diâmetro da lente ( $D$ ):

$$N = \frac{f}{D} \quad [2]$$

A Figura 2 demonstra uma representação de diversas lentes, diminuindo o diâmetro para uma distância focal constante, ou seja, incrementando a razão focal.



**Figura 2 – Incremento de raio focal.**

Para um tempo de exposição constante, um aumento na razão focal resultaria num aumento de EV.

A razão focal também determina a profundidade do campo, em outras palavras, o trecho de profundidade do espaço que será capturado fielmente.

### 2.1.4. A captura de imagem pelo EV

Câmeras convencionais são limitadas a um EV fixo por foto. Diante de uma paisagem, regulam seu tempo de exposição e raio focal para um EV médio, um equilíbrio para tudo ficar razoavelmente visível. Algumas, ainda, permitem a escolha de qual região da imagem possui maior importância, e com base nisso, ajusta o EV para dar preferência para aquela área. Em ambas as maneiras, por conta das câmeras optarem por um único EV, as fotografias são de *Low Dynamic Range* (pequeno alcance dinâmico), LDR, ou ainda de *Standard Dynamic Range* (alcance dinâmico padrão), SDR.

Com os conhecimentos adquiridos, pode-se voltar à definição de HDR apresentada no início deste documento. Sintetizando, uma imagem em HDR seria, então, uma foto ou sequência de fotos em que cada imagem individual é capturada em diferentes EVs, de forma a manter uma boa qualidade e baixo contraste ao longo da imagem independente de diversos focos luminosos. A Figura 3 ilustra uma paisagem capturada em HDR.

Nota-se a boa qualidade tanto na vegetação quanto no céu claro, resultando em uma bela e detalhada representação da paisagem em questão. A imagem foi capturada utilizando uma câmera fotográfica convencional.



**FIGURA 3 - Paisagem capturada em HDR. Créditos de imagem: Winter Wonderland II de Philip Klinger.**

## 2.2. Procedimento HDR

Imagens em HDR que se comparam ou até ultrapassam o alcance dinâmico do olho humano podem ser obtidas por meio de sensores de câmeras comuns. A imagem HDR é resultado da superposição de diversas imagens LDR ou SDR em uma mesma cena, cada uma dessas com um EV diferente.

Adquirindo imagens para que se possa sintetizar uma imagem com grande alcance dinâmico, precisa-se primeiro obter múltiplas capturas, com baixo alcance dinâmico e com tempos de exposição diferentes, de uma mesma cena. Não há um número máximo de fotos que possam ser utilizadas, pois a quantidade dependerá do algoritmo a ser utilizado e, sobretudo, da diferença de luminosidade entre o ponto mais claro e o mais escuro do ambiente. Cenas com menor diferença entre esses pontos exigirão um menor número de imagens para um resultado satisfatório.

As imagens obtidas deverão estar alinhadas, ou seja, cada pixel da foto deverá representar a mesma posição espacial em todas as diferentes imagens. Caso ocorra alguma movimentação na câmera ou objetos presentes na cena, durante a captura de imagens, será necessário implementar algoritmos que compensem tais movimentos. Como as imagens a serem alinhadas não são idênticas, pois possuem EVs e luminosidades diferentes, é preciso implementar algoritmos que levem em conta tal fato.

### 2.2.1. Alinhamento de imagens

Em muitos casos, mesmo que as imagens tenham sido capturadas com uma câmera montada em um tripé, ocorrem desalinhamentos entre as capturas de diferentes tempos exposição. Para que tais imagens possam ser utilizadas posteriormente, é necessário alinhá-las. Soluções convencionais, que utilizam detecção de borda para fazer o alinhamento das imagens, tendem a falhar, com bordas aparecendo e sumindo entre os diferentes valores de exposição. Além disso, alinhamento de imagens baseado em tal método consome uma considerável quantidade de tempo, tornando inviável o processamento de imagens em tempo real. Para tanto, deve-se utilizar um método que seja independente da luminosidade da foto (tempo de exposição) e extremamente rápido (WARD, 2003).

O método sugerido por Greg Ward, em seu artigo, consiste em transformar todas as imagens a serem alinhadas em *bitmaps*, nos quais os valores dos pixels variam de 0 a 1. Para decidir o valor do pixel, é preciso, primeiro, fazer a média aritmética do valor de todos os pixels da imagem. Com tal cálculo feito, pode-se definir o valor 1 para pixels

com valores acima da média e, consequentemente, 0 para pixels abaixo da média. Como a média se ajusta, compensando as condições de luminosidade de cada imagem, podem-se obter *bitmaps* idênticos provindos de fotos com diferentes tempos de exposição e, assim, utilizar *offsets* e operações binárias de comparação.

### 2.2.2. Recuperando a curva de resposta

Com as imagens devidamente capturadas e alinhadas, é necessário descobrir o fator equivalente entre as diversas imagens.

Devido às técnicas de construção, imperfeições e processos semicondutores, o processo de transformação de luz (fótons) em corrente elétrica não é linear, o que não torna possível a conversão direta do valor dos pixels em radiância (quantidade de luz). Para isso, é preciso primeiro, obter a curva de resposta do *setup* óptico utilizado, ou seja, calcular como todos os componentes da câmera (lentes, obturador e sensor) devem comportar-se nos diversos tempos de exposição. Tal feito pode ser realizado utilizando-se ferramentas matemáticas que têm como função a posição espacial de  $N$  pixels e seus valores nos diversos tempos de exposição (LE, 2007), sendo  $N$  o número de pixels utilizados para obter a curva de resposta (Figura 4).

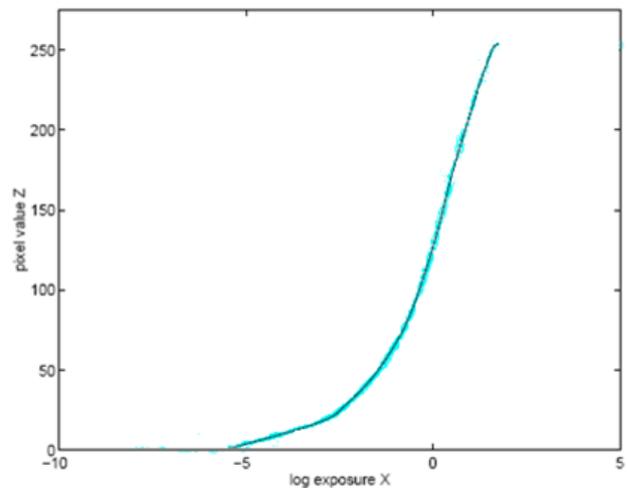


Figura 4 – Curva de resposta recuperada.

Existem várias maneiras de se recuperar a curva de resposta de um *setup* óptico. Apesar de todas levarem em consideração o que já foi dito anteriormente, existem algoritmos mais complexos e robustos que fazem cálculos ponderados dando maior peso a pixels que se encaixem no meio da curva de resposta e ignorando os pixels saturados (DEBEVEC, 1997). Com o auxílio da Equação 3, deduzida por Debevec em sua pesquisa, é possível descobrir o valor da irradiância ( $E$ ) e, consequentemente, obter a curva de resposta ilustrada anteriormente, com os valores de  $i$ , representando a posição espacial do pixel,  $j$

representando o tempo de exposição,  $Z$  representando o valor digital do pixel e  $X$ , o equivalente ao valor de exposição (EV).

$$\ln(f^{-1}(Z_{ij})) = \ln(E_i) + \ln(\Delta t_j) \quad [3]$$

### 2.2.3. Radiance Map

Com as imagens capturadas alinhadas e a curva de exposição devidamente recuperada, o próximo passo é sintetizar, mediante algoritmos matemáticos, o *radiance map* (mapa de radiância). Trata-se de uma imagem intermediária no processo de obtenção da imagem final, cujo valor dos pixels representa a quantidade de luz emitida ou refletida na direção do sensor da câmera (E).

Para realizar tal feito pode-se utilizar uma variação da Equação 4 para obtenção da resposta sugerida por Debevec, onde  $P$  é o número de fotografias,  $i$  representa a posição espacial do pixel,  $j$  representa o tempo de exposição,  $Z$  representa o valor digital do pixel,  $E$  é a irradiância. Tal variação leva em conta o tempo de exposição, a posição espacial dos pixels e uma função ponderada que dá maior contribuição para os pixels no meio da curva de resposta. Isto torna esta equação mais robusta conforme mais tempos de exposição são utilizados, além disso, a função ponderada ajuda a reduzir não linearidades e ruídos na imagem.

$$E_i = \frac{\sum_{j=1}^P \omega(Z_{ij}) \cdot (g(Z_{ij}) - \ln(\Delta t_j))}{\sum_{j=1}^P \omega(Z_{ij})} \quad [4]$$

O mapa de radiância (Figura 5a) por si só já é uma imagem em HDR, porém para que se possa utilizá-lo é preciso primeiro que o mesmo seja readequado aos monitores convencionais por *via* de algoritmos chamados de *tone map* (Figura 5b).

### 2.2.4. Tone Map

Ao contrário de todos os processos vistos até agora que buscam resultados exatos, o algoritmo de *tone map* trabalha com a percepção humana sobre a cena em questão. Idealmente se utilizaria um monitor capaz de exibir imagens em HDR para mostrar a mesma, obtendo o maior nível de

fidelidade luminosa possível da imagem observada. Infelizmente, tais monitores ou *displays* são raros, caros e começaram a ser desenvolvidos há pouco tempo. Para que imagens e vídeos em HDR possam ser aproveitados pelo maior número de pessoas possível, é preciso que os mesmos sejam compatíveis com a grande gama de equipamentos reprodutores já existentes.

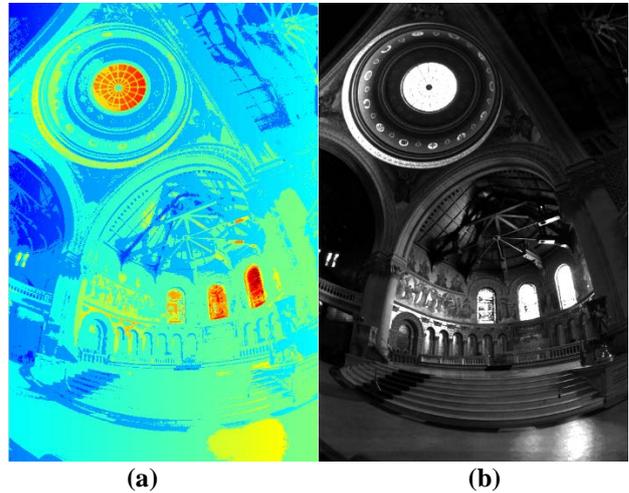


Figura 5 – (a) Exemplo de *radiance map* e (b) exemplo de foto HDR.

Para realizar tal tarefa de compatibilidade, precisa-se comprimir o range dinâmico das imagens HDR de modo que as mesmas possam ser exibidas em monitores LDR. Esse processo não busca uma alta fidelidade com os índices reais de radiância na cena e sim em parecer fiel para a percepção humana.

O principal objetivo de um algoritmo de *tone map* é usar o máximo possível do *range* dinâmico do *display* a ser utilizado e obter a maior fidelidade de cores possível. Kiser *et al.* (2012) sugerem que para obter tal fidelidade, e ainda assim utilizar o maior *range* dinâmico, é necessário um algoritmo ponderado, saturando 1% dos pixels de maior e de menor luminância, isso evitaria que áreas escuras se tornassem cinzas e áreas claras aparecerem escurecidas. Um exemplo do método descrito é apresentado na Figura 6.



FIGURA 6 – Imagens obtidas (a) com saturação de pixels (Kiser) e (b), sem saturação de pixels.

As equações 6, 7, 8, 9 e 10 determinam os valores dos pixels obtidos na imagem A, onde  $L$  é a luminância (radiância),  $L_{av}$  é a luminância média,  $L_{white}$  é o menor valor que deve ser convertido para branco; a partir de uma imagem intermediária (*radiance map*) (KISER *et al.*, 2012).

$$a = 0,18 \cdot 2^2 \cdot \frac{(B - A)}{(A + B)} \quad [6]$$

$$A = L_{max} - L_{av} \quad [7]$$

$$B = L_{av} - L_{min} \quad [8]$$

$$L(x, y) = a \cdot L(x, y) \cdot L_{av} \quad [9]$$

$$L_d(x, y) = L(x, y) \cdot \frac{1 + L(x, y) \cdot L_{white}^2}{1 + L(x, y)} \quad [10]$$

Outros desafios são encontrados quando vídeos em HDR são readequados para monitores convencionais. Devido aos parâmetros que variam das equações mostradas anteriormente, grandes mudanças nos valores de luminosidade podem ocorrer de *frame* para *frame*, criando um efeito não desejado no vídeo a ser mostrado, isto cria a necessidade de adicionar certa histerese nos parâmetros variáveis (KISER *et al.*, 2012).

### 2.2.5. OpenCV

*Open Source Computer Vision Library* (Biblioteca de Visão Computacional em Código Aberto), abreviado em OpenCV, é uma biblioteca em código aberto, sob licença BSD, completa com mais de 2.500 algoritmos na área de processamento de imagem.

A biblioteca conta com interfaces para desenvolvimento nas linguagens C, C++, Python e Java, inclusive para MATLAB. Tem suporte a sistemas operacionais como Windows, distribuições Linux, Android e Mac OS, sendo muito utilizada atualmente em grandes indústrias, como Google, Microsoft, Yahoo, entre outras.

Os algoritmos da biblioteca variam desde funções básicas para manipulação de imagens até algoritmos mais complexos como o de reconhecimento facial, identificação de objetos, criação de imagens 3D, entre outros.

Um programa OpenCV pode ser implementado em uma grande variedade de dispositivos graças à versatilidade da linguagem C. A presença de OpenCV em sistemas embarcados é de fundamental importância, principalmente no desenvolvimento de robôs, pequenos módulos de captura e processamento de imagem, sistemas de monitoramento e controle industrial.

## 3. MATERIAIS E MÉTODOS

Utilizando a biblioteca OpenCV, para a linguagem de programação C, e as imagens fornecidas por Debevec em seus estudos, pôde-se dar início aos algoritmos criados para a elaboração de uma imagem em HDR. Como escopo limitante, o algoritmo foi pensado e criado para apenas trabalhar com imagens previamente alinhadas. As etapas necessárias para o desenvolvimento dos algoritmos foram esquematizadas a partir do conhecimento adquirido na fundamentação teórica.

### 3.1. Primeiro algoritmo

O primeiro algoritmo elaborado tem como objetivo mesclar um número indefinido de imagens carregadas no mesmo e produzir uma imagem HDR capaz de ser exibida em monitores convencionais. Para isso, foi desenvolvido um programa de computador em linguagem de programação C que faria o processamento com base no diagrama de blocos exposto na Figura 7.

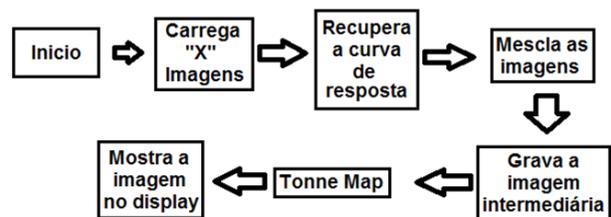
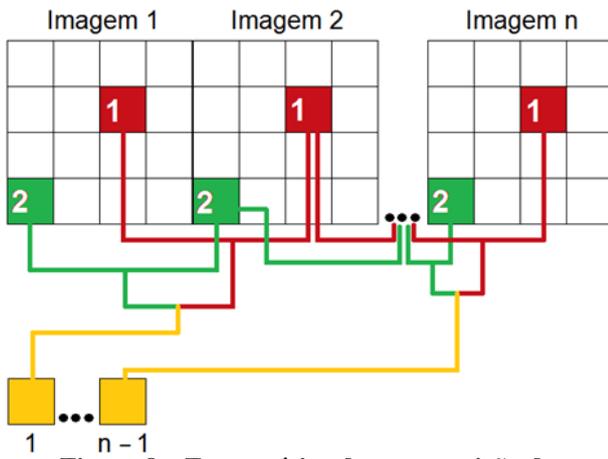


Figura 7 – Diagrama em blocos do sistema.

Verificando o diagrama de blocos pode-se notar que o algoritmo carrega todas as imagens a serem processadas na memória RAM do computador, as quais serão nomeadas “imagens fonte”. Após isso, é criada uma imagem a partir das funções disponíveis pelo OpenCV, a qual será denominada imagem intermediária. A imagem intermediária tem como função receber os resultados dos cálculos que serão realizados com base nas imagens fonte. É importante ressaltar que, para o algoritmo desenvolvido, a sequência em que as imagens são carregadas é relevante, sendo necessário o carregamento da imagem mais clara para a mais escura (maior para o menor tempo de exposição).

#### 3.1.1. Recuperando a curva de resposta

Para que pudessem ser realizadas as operações posteriores de mesclagem entre as imagens fonte, foi preciso descobrir a relação de equivalência entre elas. O método utilizado para se obter tal equivalência foi recuperar a curva de resposta do *setup* óptico utilizado para obter as imagens fonte (Figura 8).



**Figura 8 – Esquemático de superposição de pixels.**

Este método consiste em encontrar a equivalência de cada imagem, em relação à outra, a partir da razão entre pixels de posição correspondentes, como pode ser visto na representação da junção de dois pixels correspondentes de imagens diferentes na Figura 8, e, em seguida, são armazenados num vetor (em amarelo). Para fins de redução de ruído, vários pixels são selecionados para realizar esse cálculo, obtendo, assim, um valor de equivalência médio que é, então, gravado em um vetor.

É importante ressaltar que a razão entre pixels correspondentes apenas pode ser feita entre imagens com luminosidade parecida, visto que em imagens com diferença luminosa muito grande existiriam poucos pixels de posição correspondente não saturado em uma e em outra imagem.

Neste ponto, o valor de cada razão representa a relação de equivalência entre imagens adjacentes. Para que todas as imagens sejam referenciadas em uma única imagem, foi considerado o produto das razões previamente armazenadas de maneira com que a razão de equivalência fosse referenciada à última imagem fonte carregada.

### 3.1.2. Mesclando as imagens

Com a curva de resposta previamente recuperada foi possível mesclar as imagens fonte em uma única imagem intermediária. O método utilizado para realizar tal tarefa foi multiplicar cada imagem fonte por seu valor de correspondência (obtido mediante a curva de resposta) e, então, somá-las à imagem intermediária.

### 3.1.3. Tone Map

Para que a imagem intermediária possa ser exibida em monitores convencionais, é preciso primeiro readequá-la para isso. Como os monitores convencionais comportam apenas 8 bits por cor, o valor dos pixels presentes na imagem intermediária precisam ficar contidos entre 0 e 255.

Para manter os valores da imagem intermediária dentro dos limites impostos, foi realizada uma varredura a procura do menor e do maior valor de pixel. Em seguida cada pixel teve seu valor alterado pela Equação 11, onde  $P$  representa o valor do pixel,  $P_{max}$  representa o maior valor de pixel encontrado e  $P_{min}$  o menor valor de pixel encontrado.

$$P = \frac{255 \cdot (P - P_{min})}{(P_{max} - P_{min})} \quad [11]$$

## 3.2. Algoritmo modificado

O segundo algoritmo tem os mesmos objetivos do primeiro algoritmo. O mesmo apresenta modificações na parte de obtenção da curva de resposta, visando dar maior peso aos tons e luminosidades medianas.

### 3.2.1. Curva de resposta

Visto que as imagens resultantes do primeiro algoritmo são muito escuras, foi preciso reduzir a distância entre as razões de equivalência, para que um maior peso fosse dado para os tons médios, de modo que as imagens com maior luminosidade ficassem com um maior peso em relação às imagens com pouca luminosidade.

Isso foi alcançado repetindo-se o processo de recuperação de curva de resposta realizado do primeiro algoritmo, porém, desta vez, o valor da razão ficou referenciado em imagens adjacentes em vez de referenciados à última imagem carregada.

## 4. RESULTADOS

Os resultados obtidos foram adquiridos carregando-se 11 imagens, previamente alinhadas, nos dois algoritmos desenvolvidos. Tais imagens possuem tempos de exposição variando de 32 a 0,03125 segundos e foram disponibilizadas por Debevec em seus estudos, podendo ser observadas na Figura 9.

### 4.1. Primeiro algoritmo

Carregando as imagens fonte no primeiro algoritmo, foi obtida a curva de resposta presente na Figura 10. Nesta plotagem da curva de resposta fica claro observar que imagens mais escuras serão multiplicadas por um número maior, e em consequência disso terão uma maior contribuição na imagem final.

Pode-se notar neste gráfico a presença de uma região linear, abrangendo a maior parte do gráfico e de uma região de saturação onde a razão de equivalência tende a ficar constante. Com a curva de resposta do primeiro algoritmo e todos os processos descritos na metodologia realizados, obteve-se a imagem presente na Figura 11.

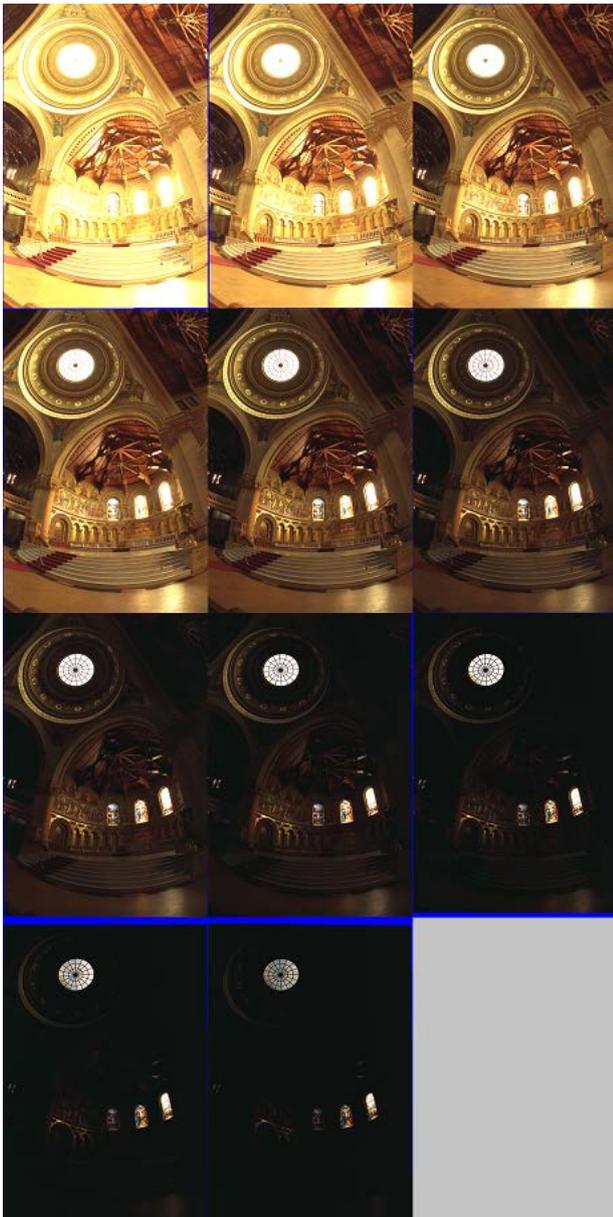


Figura 9 – Compilado de imagens utilizadas.

que áreas muito claras, como as janelas e vitrais, podem ser identificadas como áreas de menor luminosidade. Apesar disso, a imagem ainda se encontra muito escura, fazendo com que as áreas de baixa luminosidade não tenham um grau de visibilidade satisfatório.

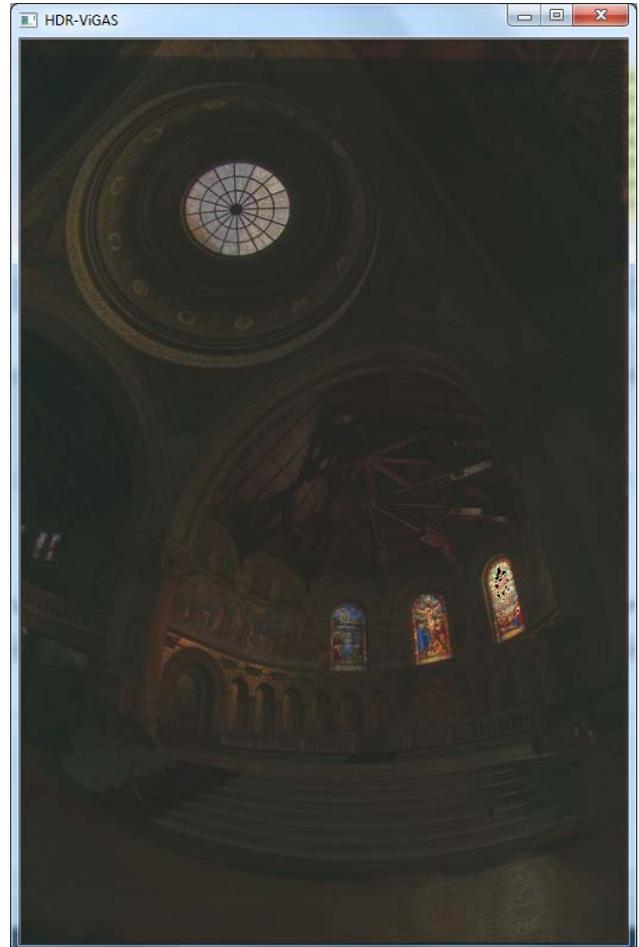


Figura 11 – Resultado obtido do primeiro algoritmo.

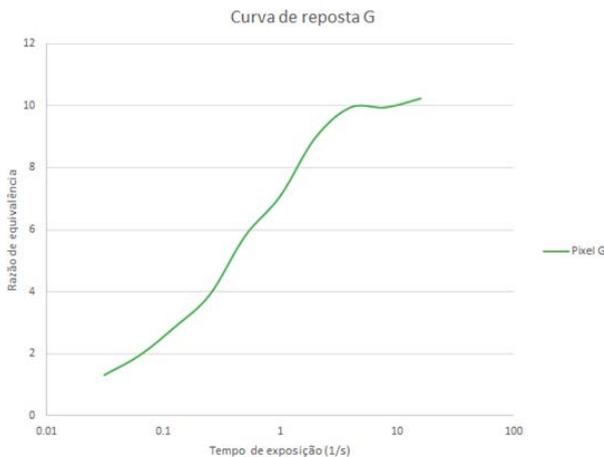


Figura 10 – Curva de resposta obtida para a cor R.

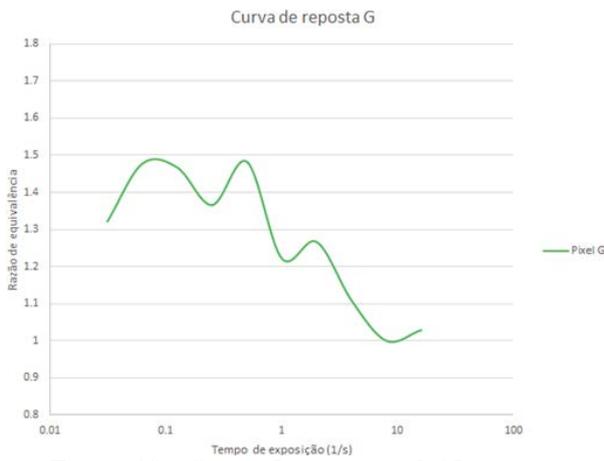
Pode-se observar na imagem da Figura 11 uma significativa melhora de alcance dinâmico. Comparando-a com as imagens fonte, pode-se notar

#### 4.2. Segundo algoritmo

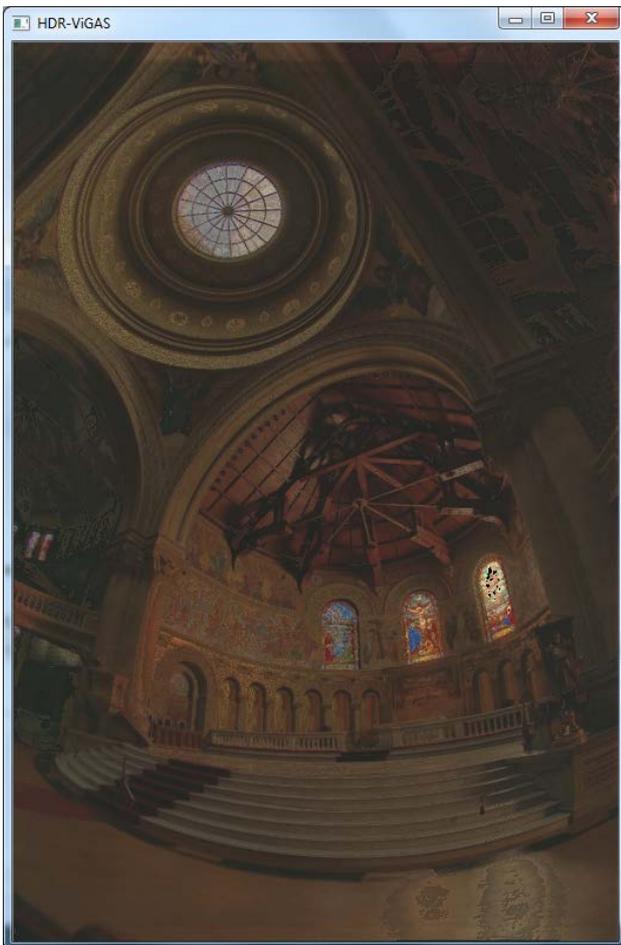
O segundo algoritmo, que tem como objetivo uma melhor visibilidade nos tons médios, teve as mesmas imagens fontes carregadas e produziu a curva de resposta que pode ser analisada na Figura 12.

Observando essa plotagem e comparando-a com a plotagem presente no primeiro algoritmo, podem-se perceber algumas diferenças. A primeira é a tendência da curva, que atribui menor peso para imagens mais escuras, representadas pela parte final do gráfico. A segunda diferença é a falta de uma região linear, caracterizada pelos picos e vales entre imagens.

Os processos descritos na metodologia e esta curva de resposta produziram a imagem que pode ser observada na Figura 13.



**Figura 12 – Curva de resposta obtida para a cor G.**



**Figura 13 – Resultado obtido do segundo algoritmo.**

Analisando essa imagem pode-se perceber uma significativa melhora de visibilidade comparando com a imagem produzida pelo primeiro algoritmo ou até imagens fonte. É possível visualizar nessa imagem áreas muito claras, como as janelas e vitrais, assim como áreas muito escuras, presentes nas laterais da imagem.

Porém, pode-se notar um aumento na quantidade de ruído presente na imagem, principalmente em áreas como os arredores da

janela superior, zona inferior dos degraus (canto direito) e telhado superior direito. Tais imperfeições ficam ainda mais claras quando comparadas à imagem obtida por Debevec, Figura 14.



**Figura 14 – Resultado obtidos pela pesquisa de Debevec.**

Esta imagem apresenta uma maior nitidez e fidelidade às cores e aos tons reais da cena, assim como uma imagem globalmente mais clara, porém, a mesma apresenta uma baixa visibilidade nas áreas mais claras da imagem, fazendo com que os vitrais e as janelas ficassem totalmente brancos e saturados.

## 5. CONCLUSÕES

Com a aplicação do algoritmo computacional desenvolvido neste trabalho e a consequente geração de uma imagem em HDR, obtiveram-se ótimos resultados para áreas de alta luminosidade, inclusive superior aos pontos luminosos nos resultados de Debevec. Em contrapartida, notou-se a ocorrência de ruídos nas imagens obtidas, além da apresentação dos cantos escuros com pouca visibilidade em relação às imagens do Debevec.

Os desafios e dificuldades encontrados nesses algoritmos serviram como um meio de aprendizagem, possibilitando diversas melhorias para os trabalhos futuros. Encontrando maneiras

mais fiéis e menos suscetíveis a ruídos para capturar a curva de resposta da imagem, pode-se diminuir a quantidade de ruído global na imagem. Implementando outras técnicas de *tone map*, que não sejam lineares, pode-se clarear as zonas mais escuras da imagem, fazendo com que a cena fique globalmente mais clara.

Os próximos passos são realizar a implementação do algoritmo em tempo real, capaz de processar as imagens providas de uma câmera, alinhá-las, mesclá-las e mostrá-las em um *display*. O algoritmo deverá ser executado por um processador ARM com aceleração por *hardware* feito com uma FPGA.

#### **Agradecimentos**

Os autores agradecem a Pró-Reitoria de Pesquisa, Pós-Graduação e Inovações (PROPI) do IFSC, pelo incentivo financeiro ao projeto ViGAS através do edital de N° 19/2014.

#### **REFERÊNCIAS**

DEBEVEC, Paul E.; MALIK, Jitendra. *Recovering high dynamic range radiance maps from photographs*. In: **ACM SIGGRAPH 2008 classes**. ACM, 2008. p. 31. Disponível em <<http://www.pauldebevec.com/Research/HDR/debevec-siggraph97.pdf>>. Acesso em 14 out. 2014.

KISER, C.; REINHARD, E.; TOCCI, M.; TOCCI, N. *Real time automated tone mapping system for HDR video*. In: **IEEE International Conference on Image Processing**. 2012. p. 2749-2752.

LE, Qiyue; XI, Zhu. *Recovering high dynamic range radiance and localized tone-mapping algorithm*. *Project Report*, 2007. Disponível em <[http://www.sfu.ca/~zhuxiz/pdf/hdr\\_report.pdf](http://www.sfu.ca/~zhuxiz/pdf/hdr_report.pdf)> . Acessado em 19 set. 2014.

WARD, Greg. *Fast, robust image registration for compositing high dynamic range photographs from hand-held exposures*. **Journal of graphics tools**, v. 8, n. 2, p. 17-30, 2003. Disponível em: <<http://pages.cs.wisc.edu/~lizhang/courses/cs766-2008f/projects/hdr/jgtpap2.pdf>>. Acesso em 19 set. 2014.