

Utilização de técnicas de preenchimento para correção de reflexos luminosos em imagens

Edson Gabriel Andreoli Ferreira¹, Alan Salvany Felinto¹

¹Departamento de Computação – Universidade Estadual de Londrina (UEL)
Caixa Postal 10.011 – CEP 86057-970 – Londrina – PR – Brasil

egabriel@uel.br, alan@uel.br

Abstract. *The human being is very visual, small details such as a light reflection in a photo can cause its sharpness as an image, in most cases these problems do not cause big problems, however there is the occasion that errors like these cannot be ignored, the problem dealt with here is how to correct light reflections by getting as close as possible to the reflection-free image.*

Resumo. *O ser humano é muito visual, detalhes pequenos como um reflexo luminoso em uma foto podem acarretar na perda da qualidade da imagem, na maioria das vezes esses problemas não causam grandes problemas, porém a ocasião que erros como esses não podem ser ignorados, o problema tratado aqui é como corrigir reflexos luminosos deixando as mais próximo possível da imagem sem reflexo.*

1. Introdução

Ao longo do tempo a evolução da fotografia desde a câmera escura até era digital houve um grande progresso na qualidade de imagem, notando que a importância de uma boa imagem nos tempos de hoje possui grande relevância, levantando até onde vai à qualidade dessas imagens, a mudança que tivemos no uso delas foi enorme, de fotos do espaço à imagem de micro-organismos nos tempos atuais, mostra se como o mundo está dependente delas, exames médicos, periciais, fotos. Há hoje no mercado, diversas ferramentas de tratamento de imagens, até tratamento de borrões e os chamados efeitos de fotos que se popularizam com uso dos smartphones.

A perda de qualidade da imagem podem ocasionar dependendo da situação a sua errada interpretação, o ponto crucial aqui é que a falta de uma parte da imagem causada por um fator que possa ser corrigido deve ser notado, reflexos luminosos numa foto podem ou não ter impacto dependendo daquilo que esta sendo feito, em uma foto de paisagem ou artística esse problema pode ser ignorado facilmente, porém ocasiões onde a perda mesmo pequena da informação contida na imagem podem acarretar tomadas de decisões erradas, um exame medico, uma foto publicitaria entre outras.

Achar a melhor maneira de como consertar esses reflexos luminosos nas imagens é o objetivo desse trabalho, encontrar as técnicas que possam consertar esses reflexos luminosos nas imagens e a partir disso com ajuda de métricas comparar a eficácia e os possíveis ajustes serão pontos debatidos nesse trabalho de conclusão de curso.

2. Fundamentação Teórico-Metodológica e Estado da Arte

2.1. Base de dados

Inicialmente obtenção de imagens será a partir da base de dados Yale face database[7] nela encontram-se varias imagens de faces humanas, e também varias formas da mesma imagem como imagem com luminosidade, sem luminosidade, imagem sem opacidade entre outros. A outra base também será usada é o Flickr Material Database[8], aqui teremos como base, diversas imagens de vários objetos do cotidiano como Facas, objetos plásticos e muitos outros e também com vários graus de luminosidade.

Além dessas bases de dados citadas, também serão utilizados bases criadas para o projeto, e novas bases de dados serão encontradas com decorrer do projeto.

2.2. OpenCV

OpenCV(Open source Computer Vision) é uma biblioteca de uso livre, utilizada para análise de imagens, vídeos, detecção e reconhecimento facial, entre outros. Possui em diversas funções, desenvolvida primeiramente para C++, porém hoje possui suporte para diversas linguagens de programação Java,Python,Ruby e entre outras.OpenCV dentro da sua biblioteca possui dois algoritmos de preenchimento[6].

2.2.1. Floodfill

Esse algoritmo de preenchimento, um ponto é selecionado dentro da imagem, toda área conectada a esse ponto de mesma cor é preenchida com atributo correspondente, a conectividade é determinada pela cor/brilho dos pixels vizinhos, possui função semelhante à ferramenta balde em programas de pinturas [3].

2.2.2. Boundary fill

Algoritmo de preenchimento da biblioteca do OpenCV e que trabalha com preenchimento por limite, assim como floodfill um ponto da imagem é selecionado, a partir do ponto é disparado para direções, N,S,L,O ou N,S,L,O,NO,SO,NE,NO, escolhendo a cor, ele percorre a imagem, substituído a cor ao até chegar o limite decidido, porém esse algoritmo só funciona se a cor de escolhida for diferente da cor limite [1].

2.3. Inteligencia artificial

Para preencher esses ‘buracos’ a utilização de algoritmos envolvendo Inteligencia artificial serão necessários, abaixo serão descritos as técnicas de preenchimento envolvendo IA para solução do problema.

Nesta técnica o autor cria-se uma lacuna na imagem, por exemplo, um buraco branco com a aplicação da Inteligência artificial cria uma estrutura que aborda a região da imagem ausente no caso, com base no restante da imagem cria um modo de preenchimento da imagem para a lacuna[9].

2.4. Métricas

Com a aplicação dos preenchimentos que serão utilizados nas imagens, Para avaliação dessas técnicas será usado métricas, para este trabalho usaremos a comparação da imagem com sua imagem referencia, conhecida como referência completa as seguintes métricas serão utilizadas abaixo:

2.4.1. Mean Squared Error - MSE

MSE ou Erro quadrático médio em português, é uma das métrica mais simples e utilizada, tem como base a estatística a ideia aqui é comparar dois sinais, o original e a imagem modificada criando uma pontuação que representa a similaridade das imagens sendo que quanto menor o erro maior chance de imagens serem parecidas[2].

2.4.2. Peak Signal-to-Noise Ratio - PSNR

PSNR ou Relação sinal-ruído de pico em português, nesta métrica é feito um cálculo entre duas imagens pelo seu sinal de ruído, esse sinal quanto maior melhor será, auxiliado pelo MSE é usado como métrica de comportamento de imagens [5].

2.4.3. Structural similarity - SSIM

O SSIM ou Similaridade Estrutural é uma métrica que qualifica a degradação da imagem a partir do seu índice usado para comparar a semelhança entre duas imagens, ou seja utiliza da imagens como referencia onde tem se a imagem sem defeito e a nova imagem. Esse modelo de comparação tem como base nas estruturas, luminância e contraste da imagem [4].

3. Objetivos

O objetivo principal é encontrar maneiras de preencher essas lacunas deixado pelos reflexos luminosos, além de procurar quais são as melhores técnicas que trara os resultados mais satisfatório dependendo do tipo de imagem, também identificar o quão é possível deixar a imagem próxima da sua nitidez, definindo qual é o limite de correção.

4. Procedimentos metodológicos/Métodos e técnicas

Para elaboração do projeto, será feito primeiramente um levantamento da base de imagens, a partir disso uma pesquisa bibliográfica de trabalhos similares e algoritmos de preenchimento de regiões será feita, definindo os algoritmos um levantamento de métricas de qualidade de imagens será realizado e por último os resultados das comparações dos algoritmos de preenchimento será executado.

Esse Trabalho será desenvolvido com base no OpenCV, python e a utilização de Deep Learning. O acompanhamento do TCC será por reuniões semanais e utilizará da metodologia de desenvolvimento Ágil, Scrum com o apoio da ferramenta web Trello.

4.1. Cronograma de Execução

Os seguintes passos serão realizados para elaboração do projeto:

A) Levantamento de base de imagens que possuem imagens originais e segmentadas.

B) Levantamento bibliográfico e escolha de algoritmo de preenchimento de regiões.

C) Levantamento bibliográfico e escolha de métricas de qualidades de imagens.

D) Comparação dos algoritmos de preenchimento de região com base nas métricas de qualidade de imagens escolhidas.

E) Escrita do TCC.

Com base nas atividades descritas no procedimento metodológico segue o cronograma:

Tabela 1. Cronograma de Execução

	set	out	nov	dez	jan	fev	abr	mai
A	x							
B	x	x						
C		x	x					
D				x	x	x	x	
E					x	x	x	x

5. Contribuições e/ou Resultados esperados

Neste trabalho espera-se encontrar uma solução ideal para preenchimento dos reflexos luminosos em imagens, com utilização de técnicas ajustadas ao problema, almeja que os resultados encontrados nesse trabalho possam ser de ajuda a problemas similares.

6. Espaço para assinaturas

Londrina, 13 de setembro de 2021.



Aluno



Prof Dr. Alan Salvany Felinto
Orientador

Referências

- [1] Anjali Agarwal. Boundary fill algorithm. <https://www.geeksforgeeks.org/boundary-fill-algorithm/>. Acessado: 2020-09-06.
- [2] Rafael Borges de Souza. Métricas em processamento de imagens. ., 10 de Agosto 2020.

- [3] FreeCodeCamp. Flood fill algorithm explained. <https://www.freecodecamp.org/news/flood-fill-algorithm-explained>. Acessado: 2020-09-05.
- [4] Shiqi Wang Keyan Ding, Kede Ma and Eero P. Simoncelli. Comparison of full-reference image quality models for optimization of image processing systems. *International Journal of Computer Vision*, 21 de Janeiro 2021.
- [5] MathWorks. Psnr. <https://www.mathworks.com/help/vision/ref/psnr.html>. Acessado: 2020-09-07.
- [6] OpenCV. *The OpenCV Reference Manual*, 2.4.13.7 edition, Abril 2014.
- [7] D. Kriegman P. B, J. H. Recognition using class specific linear projection. In *Ó IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages pp. 711–720. Publishing Press, Julho 1997.
- [8] Lavanya Sharan, Ruth Rosenholtz, and Edward H. Adelson. Accuracy and speed of material categorization in real-world images. *Journal of Vision*, 14(10), 2014.
- [9] Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Julho 2017.