



Jáder Louis de Souza Gonçalves¹

Nataly Bonfim Tobias²

Orientador: Dr. Lucas Marques de Cunha³

INTRODUÇÃO

Na era atual da tecnologia digital, o processamento de imagens desempenha um papel crucial em uma variedade de campos, variando do entretenimento à ciência e tecnologia. Reconhecendo essa necessidade, o projeto "Lumina Pixel Studio" foi desenvolvido como um software educacional de testes para processamento de imagens. O objetivo principal deste trabalho é desenvolver o aplicativo Lumina Pixel Studio, que contém a integração de várias funcionalidades essenciais para o processamento de imagens.

REVISÃO DE LITERATURA

2.1 O que é Processamento de Imagens

O Processamento de Imagens é um campo que se dedica à modificação e análise de imagens através do uso de computadores. Conforme explicado por Chavéz (2013), este campo envolve a alteração das informações contidas nas imagens sob diversos aspectos. O objetivo pode ser tanto a produção de uma nova imagem quanto a extração de informações específicas a partir da imagem original. Este processo é realizado através de um conjunto de técnicas que permitem capturar, representar e transformar imagens com o auxílio de tecnologias computacionais.

2.1.2 Transformação de Intensidade

A Transformação de Intensidade, segundo Ponti Jr. (2013), é um aspecto fundamental do processamento de imagens digitais. Essa técnica envolve a manipulação dos valores de intensidade (ou níveis de cinza) em uma imagem para alcançar resultados específicos, como realçar características, corrigir contraste, ou modificar a aparência geral da imagem.

2.1.3 Detecção de bordas

A detecção de bordas, conforme descrito por Seara e Elizandro (2013), é uma técnica crucial no processamento de imagens digitais. Esta técnica é focada na identificação de pontos em uma imagem onde ocorre uma mudança brusca ou uma descontinuidade nas intensidades.

2.1.4 Transformação geométrica

A Transformação Geométrica, segundo Mezaadri e Pissini (2016), é um aspecto importante no campo do processamento de imagens. Esta técnica envolve a alteração da geometria de uma imagem, incluindo operações como rotação, escala, translação, e distorção.

2.1.5 Operações Aritméticas

As operações aritméticas no processamento de imagens, conforme descrito por Chávez (2013), são procedimentos fundamentais que envolvem a manipulação de imagens através de operações algébricas básicas como soma, subtração, multiplicação e divisão, aplicadas pixel a pixel. Estas operações permitem a criação de uma nova imagem $C(x,y)$ a partir da combinação de duas imagens de entrada $A(x,y)$ e $B(x,y)$.

2.1.5.2 Convolução

Esta é uma operação matemática que combina duas funções para produzir uma terceira função. No contexto da pesquisa de Chávez (2013), processamento de imagens, a convolução é utilizada para aplicar um filtro sobre uma imagem. Isso é feito "convoluindo" a imagem com um kernel ou máscara, que é uma pequena matriz usada para aplicar efeitos como desfoque, nitidez, realce de bordas, entre outros.

¹ Jáder Louis de Souza Gonçalves, Ciência da Computação, UNIR, jaderlouis@proton.me.

² Nataly Bonfim Tobias, Ciência da Computação, UNIR. nataly.tobias11@gmail.com

³ Orientador: Dr. Lucas Marques de Cunha, UNIR, lucas.marques@unir.br.

No projeto Lumina Pixel Studio, foi empregada uma abordagem de pesquisa Aplicada e Bibliográfica, iniciando com estudos sobre as bibliotecas relevantes (Numpy, OpenCV, PIL) para fundamentar o trabalho com as funções de processamento de imagem. O desenvolvimento inicial das funções ocorreu em Octave, testando-as antes de transpô-las para Python, e a interface de usuário (front-end) foi primeiramente criada com Tkinter, mas devido à insatisfação com a estética, migrou-se para a Tkinter Custom, melhorando o resultado final. Este processo, documentado no GitHub, envolveu cerca de dois meses para as funções de processamento de imagem e um mês para o desenvolvimento do front-end, destacando-se o espaço para melhorias futuras, limitado pelo prazo do projeto.

4. Materiais e Métodos

O módulo `imageProcessor.py` do projeto Lumina Pixel Studio abarca uma ampla gama de funções para processamento de imagem, desenvolvidas inicialmente sem o auxílio de bibliotecas externas, para mais tarde integrar ferramentas como NumPy e OpenCV visando otimização de desempenho. Este módulo permite manipulações como redimensionamento, mesclagem através de dissolve cruzado, inversão de cores para criar negativos, ajustes de contraste, limiarização para conversão em imagens binárias, correção gama, transformações logarítmicas, melhorias de contraste via ajustes de histograma, transformações geométricas incluindo rotação, cisalhamento, e reflexão, distorções específicas, warping baseado em campos de deslocamento, suavização por filtros de mediana e média, realce de bordas com Sobel e técnicas de convolução para amplificação de detalhes e texturas. Este conjunto de funcionalidades destaca a versatilidade e profundidade do módulo para aplicações de edição e análise de imagens.

5. Resultados e Discussões

Devido ao extenso período de desenvolvimento deste projeto, é possível

categorizar os desafios enfrentados em três tipos: problemas crônicos de desenvolvimento, questões relacionadas ao processamento de imagens e dificuldades de portabilidade.

5.1 Problemas Crônicos de Desenvolvimento

No desenvolvimento do projeto Lumina Pixel Studio, optou-se por uma abordagem de desktop devido à demanda por alta performance e capacidade de processamento, enfrentando desafios crônicos como problemas de desempenho e deficiências no núcleo do aplicativo. A falta de portabilidade inicial e a dependência de bibliotecas externas, como NumPy, OpenCV e PIL, contribuíram para uma dívida técnica, resultando em uma interface visual pouco atraente e falhas de funcionamento. Após testes de estresse, decidiu-se migrar o front-end para uma biblioteca mais moderna, planejando essa mudança e a execução manual das funções para uma fase posterior do desenvolvimento, após a implementação completa das funcionalidades.

5.2 Problemas de Processamento de Imagens

O Lumina Pixel Studio enfrentou desafios significativos ao longo de seu desenvolvimento, categorizados em problemas crônicos de desenvolvimento, desafios de processamento de imagem e dificuldades de portabilidade. Optou-se pelo desenvolvimento para desktop devido à performance e capacidade de processamento, apesar das dificuldades iniciais com desempenho, interface visual e falhas de funcionamento. A transição para bibliotecas modernas foi adiada até a implementação completa das funcionalidades. No processamento de imagens, a transposição de funções do Octave para Python revelou complexidades, especialmente em operações como convolução e warping. Desafios particulares incluíram a eliminação de 'dead pixels' em rotações de imagem, o que demandou o desenvolvimento de um método próprio de interpolação bilinear, aumentando a complexidade computacional e impactando negativamente o desempenho. A decisão de reescrever funções sem bibliotecas externas aumentou ainda mais o tempo de processamento, levando a operações que podiam consumir mais de 10 minutos.

5.3 Resolução dos Problemas

Após desenvolver funções internamente, o projeto avançou para um novo

front-end, superando facilmente essa etapa. Desafios de desempenho foram abordados com otimizações, incluindo threading, algoritmos de cache, conversão de imagens para escala de cinza ou binarização, e redução de resolução. Embora outras estratégias como compressão de imagens fossem contempladas, a prioridade foi aprimorar funcionalidades, atingindo um desempenho satisfatório com planos para futuras melhorias.

6. CONCLUSÃO

O projeto Lumina Pixel Studio proporcionou uma experiência rica e detalhada em processamento de imagens digitais, abordando desafios de desenvolvimento, processamento específico de imagens e portabilidade. Revelou a importância de decisões estratégicas no ambiente de desenvolvimento, a complexidade de adaptar algoritmos de um ambiente de programação para outro e a necessidade de gerenciar a dívida técnica desde cedo. A prática em algoritmos de redimensionamento, mesclagem, transformações geométricas e técnicas de realce contribuiu para um entendimento profundo do campo. Esta vivência enriqueceu o conhecimento teórico e prático, preparando o terreno para futuras pesquisas e projetos em processamento de imagens, além de aprimorar habilidades em solução de problemas e otimização de algoritmos.

REFERÊNCIAS

CHÁVEZ, G. **Operações Algébricas e Lógicas**. 2013b. UFOP 2013. Disponível em:

<http://www.decom.ufop.br/guillermo/BCC326/slides/Processamento-de-Imagens-Operacoes-Algebrica.pdf>. Acesso em: 29 jan. 2024.

SEARA, D.; ELIZANDRO, G. **Encontrando a Linha Divisória: Detecção de Bordas**. 2013. - Inf, UFSC 2013. Disponível em:

<https://www.inf.ufsc.br/~aldo.vw/visao/bordas.pdf>. Acesso em: 29 jan. 2024.

MEZADRI, F.; PISSINI, M. **TRANSFORMAÇÕES GEOMÉTRICAS**. 2016.

UniCamp 2016. Disponível em:

https://www.ime.unicamp.br/sites/default/files/lem/material/geometria_das_transformacoes.pdf. Acesso em: 29 jan. 2024.

CHÁVEZ, G. **Fundamentos**. 2013. decom.ufo. Disponível em:

<http://www.decom.ufop.br/guillermo/BCC326/slides/Processamento-de-Imagens->

Processamento de Imagens - Universidade Federal de Rondônia.

Fundamentos.pdf>. Acesso em: 29 jan. 2024.

PONTI JR., M. **Realce de imagens parte1: operações pontuais**. Disponível em:
<http://wiki.icmc.usp.br/images/5/5e/DIP_02_Realce_p1.pdf>. Acesso em: 29 jan.
2024.