

VITOR HUGO FAZOLI DA SILVA

DESENVOLVIMENTO DE FERRAMENTA PARA
ILUMINAÇÃO EM IMAGENS BIDIMENSIONAIS DE
MÉDIA E BAIXA RESOLUÇÃO

Orientador: Prof Dr.Gabriel
Marcelino Alves

SÃO JOÃO DA BOA VISTA

2023

SUMÁRIO

1	INTRODUÇÃO	5
1.1	Objetivos	6
1.1.1	Objetivo Geral	6
1.1.2	Objetivos Específicos	6

2	REVISÃO DA LITERATURA	7
2.1	Técnicas de segmentação de imagens	7
2.2	<i>Layer-Based Segmentation</i>	7
2.3	<i>Block-Based Segmentation</i>	7
2.3.1	Métodos Baseados em Bordas (Edge-Based Methods)	8
2.3.1.1	Detecção de Bordas Clássica	8
2.3.1.2	Métodos Suaves de Detecção de Bordas	8
2.3.2	Métodos Baseados em Regiões (Region-Based Methods)	8
2.3.2.1	Crescimento de Regiões	9
2.3.2.2	Clustering	9
2.3.2.3	Técnica de Divisão e Fusão	9
2.3.2.4	Corte Normalizado	9
2.4	Inteligencia artificial	9
2.5	Segment Anything	9
2.6	Trabalhos Correlatos	9
2.6.1	Segmentation by grouping junctions	10
3	METODOLOGIA	11
3.1	Levantamento de Requisitos	11
3.2	Estudo: Python e suas bibliotecas	12
3.2.1	PyTorch	12
3.2.2	Scikit-image	13
3.2.3	OpenCV	13
3.2.4	Numpy	13
3.3	Estudo: Inteligência Artificial	13
3.3.1	Segmentação por máscara	14
3.3.2	Segmentação por pontos de clique	14
3.3.3	Segmentação por caixa de delimitação	14
3.3.4	Segmentação por texto	14
3.3.5	Segmentação interativa	14
3.3.6	Segmentação automática	14
3.4	Procedimentos e Técnicas	15
3.5	Avaliar vantagens e limitações das ferramentas abordadas	16
3.6	Modelagem da ferramenta	17
3.7	Justificativa das Tecnologias a serem adotadas	18
3.8	Avaliação Qualitativa	19
3.9	Objetivo da Metodologia	19
3.10	Design da Pesquisa	19
3.11	Amostragem	19
3.12	Procedimentos Analíticos	19

3.13	Critérios de Validação e Confiabilidade	19
3.14	Aspectos Éticos	19
3.15	Cronograma e Recursos	19
3.16	Limitações	19
4	ANÁLISE DOS RESULTADOS PRELIMINARES	20
4.1	Cronograma do Trabalho	20
5	CONCLUSÕES PARCIAIS	21
	REFERÊNCIAS	22

RESUMO

A iluminação é de grande importância para pinturas e esboços digitais nos dias de hoje, pois fornece a percepção de volume ao ambiente. Porém, a iluminação ainda é difícil de replicar em uma imagem digital, pois representa algo muito diferente de um ambiente tridimensional, ainda mais se usarmos uma imagem em baixa resolução. Nesse sentido, O objetivo do trabalho é simular a iluminação em imagens digitais de média e baixa resolução por meio do desenvolvimento de uma ferramenta. Para se alcançar o objetivo, pretende-se que a ferramenta considere informações de luz em um espaço tridimensional para simular a iluminação na imagem bidimensional. Portanto o que se espera deste trabalho é que imagens de média e baixa resolução apresentem boa iluminação.

Palavras-chave: Iluminação. Algoritmo. Ambiente. Bidimensional.

1 INTRODUÇÃO

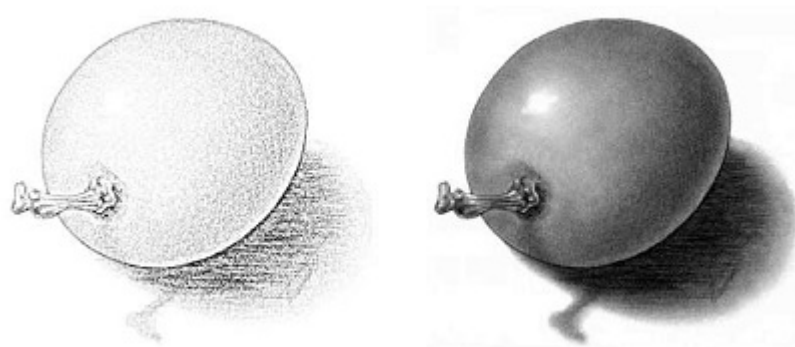
Desde que as pinturas começaram, se formaram vários estilos de arte, ainda que no princípio tudo era voltado a chegar ao realismo das pinturas, por não existir nada que pudesse retratar rostos e momentos melhor do que as artes.

No entanto, isso mudou quando surgiram as câmeras, que poderiam captar tudo de forma praticamente instantânea, com isso era o momento de se pensar que as pinturas iriam sumir, mas ela se renovou e ao invés de buscar o realismo, agora ela estava em busca de trazer novas sensações, como as pinturas abstratas, estilos como cubismo ou surrealismo que usavam de várias curvas e cores para demonstrar expressões (Cezimbra, 2021).

Esses estilos artísticos foram uma revolução no mundo da arte, como o surrealismo por exemplo, que tratava de possibilidades infinitas em suas obras, porque ali tudo era possível e o único limite era imaginação, Nesse período foram apresentadas várias obras que ao olhar não faziam sentido algum, porém era esse o princípio da ideia, pois elas eram criadas para passar uma sensação a quem olhasse, como a de liberdade, entre outros.

Atualmente a iluminação é de extrema importância para as pinturas e desenhos digitais pois é ela quem traz o volume ao ambiente, como pode ser observado na Figura 1.

Figura 1 – Duas imagens demonstrando a diferença da iluminação



(Grafitti, 2019)

Como podemos observar que na figura acima, a imagem da esquerda apresenta apenas um rascunho, sem uso das técnicas de iluminação a portanto a noção de volume e profundidade não fica muito claro, por outro lado a imagem da direita sendo algo mais agradável nas pinturas como é possível notar, além do mais várias composições de cores da luz retratada em uma imagem pode trazer vislumbres fantásticos.

Com a chegada dos primeiros modelos em três dimensões, como cubos, pirâmides e esferas entre outros, foi possível enxergar através de uma tela, um objeto qualquer que

fosse e isso mudou tudo, hoje podemos ver vários exemplos de modelos que está em nossa realidade, como a impressora 3D ou o *Billboard*¹, pois é com ela que foi possível ver a grande beleza dos modelos, é por esse motivo pelo qual atualmente a iluminação é essência para nos causar a sensação de volume, sem pensar nos algoritmos mais robustos como *ray tracing* que segundo Shirley e Morley (2008) retrata, é um algoritmo onde através de uma janela, os raios são direcionados para as imagens. As superfícies são perdidas ou atingidas por cada pixel, que é representado por um raio. Ao atingir uma superfície, o raio se refrata e continua em um novo curso, causando a formação de luzes adicionais nos arredores.

Porém para os artistas chegarem perto dos algoritmos de iluminação que criam visuais impecáveis em três dimensões, as pinturas e artes foram obrigadas a criar um volume e melhorar a iluminação, pois como a luz funcionava de forma sistemática em três dimensões, as artes puderam abrir espaço em uma variação de cores e paletas diferenciadas que trazem um dinamismo maior.

Por isso atualmente a iluminação se tornou algo tão importante para o mundo, que não importa mas o estilo que é usado, é uma parte essencial. É nesse momento em que chegamos ao cerne do problema, pois com essa importância que a iluminação tem sobre a arte e os avanços tecnológicos cada vez mais ligados a ter um *design* rápido e com eficiência, ferramentas que criam a luz em pinturas, desenhos, objetos, *sprites* ou até logos poderia impactar a arte de maneira eficiente e até mesmo para aumentar a comunidade, fazendo com que várias pessoas iniciantes que não conseguem bons resultados possam aparecer no mundo do *design*.

1.1 Objetivos

1.1.1 Objetivo Geral

O objetivo do trabalho é simular a iluminação em imagens digitais de média e baixa resolução por meio do desenvolvimento de uma ferramenta.

1.1.2 Objetivos Específicos

- Encontrar algoritmos que têm o propósito de iluminar cenas bidimensionais;
- Criar um protótipo e desenvolver a ferramenta para iluminação;
- Realizar testes na ferramenta Relight que será analisado para comparação dos resultados.

¹ apresentado em Tóquio, uma televisão imensa que pode captar imagens como se fossem 3D, criando um aspecto de profundidade pela sua curvatura

2 REVISÃO DA LITERATURA

Este capítulo será organizado de forma a se iniciar com as formas mais primitivas e os modelos de iluminação do mais simples para o mais complexo, logo após mostrar as técnicas de sombreamento e por fim mostrar as ferramentas e trabalhos correlatos ao proposto neste

2.1 Técnicas de segmentação de imagens

O autor Taneja, Ranjan e Ujjlayan (2015) mostra em seus estudos que existem diversas técnicas para a segmentação de imagem e isso está bem longe das inteligências artificiais existirem, como nos dias atuais, esses métodos por outro lado surgiram de conceitos matemáticos. Essas técnicas usufruem de cada *pixel* na imagem para entender a relação um a um com os seus adjacentes, fazendo com que possa ser medida uma similaridade entre os *pixels*, o que torna a segmentação de imagem mais do que apenas um conjunto de informações realocados em outros contextos.

Para demonstrar como essas técnicas funcionam segundo Zaitoun e Aqel (2015) é necessário sub-dividir elas em algumas categorias, a primeira bifurcação encontrada está no ponto onde a segmentação é baseada ou em camadas ou em blocos, que será mostrado em seguida:

2.2 *Layer-Based Segmentation*

O termo apresentado significa que a segmentação é baseada em camada, ou seja, segundo Zaitoun e Aqel (2015) essa técnica utiliza múltiplos detectores de objetos para criar máscaras de forma, além de analisar a aparência e a profundidade dos objetos. Isso envolve a segmentação tanto de classes quanto de instâncias, permitindo uma compreensão mais detalhada da cena.

2.3 *Block-Based Segmentation*

Os Métodos de Segmentação de Imagem Baseados em Blocos, segundo Zaitoun e Aqel (2015) são categorizados em duas propriedades: descontinuidade e similaridade, em três grupos, além de que aqui o processo de divisão dos grupos são categorizados de algumas formas como cor, continuidade, similaridade e até mesmo bordas. baseado nesse processo surgiram diversos sub categorias que iremos tratar nas próximas sub-sessões

2.3.1 Métodos Baseados em Bordas (Edge-Based Methods)

Os métodos baseados em bordas são focados em detectar discontinuidades na intensidade da imagem, ou seja, identificar transições abruptas que geralmente representam as bordas entre diferentes objetos ou regiões.

2.3.1.1 Detecção de Bordas Clássica

Entre os métodos clássicos, podemos destacar:

- **Detecção de Bordas de Roberts:** Um método simples e eficiente que utiliza operadores cruzados para calcular o gradiente espacial. Sua simplicidade o torna ideal para aplicações que exigem baixo custo computacional.
- **Detecção de Bordas de Prewitt:** Calcula a magnitude e a orientação das bordas usando uma máscara de convolução de 3x3. Embora seja mais robusto que o método de Roberts, ainda é suscetível a ruídos.
- **Detecção de Bordas de Sobel:** Aplica uma máscara 3x3 rotacionada em 90°, o que permite uma melhor suavização de ruídos enquanto calcula o gradiente das bordas. É amplamente utilizado devido à sua eficácia em detecção de bordas.

2.3.1.2 Métodos Suaves de Detecção de Bordas

Com o avanço da inteligência artificial e da computação evolutiva, surgiram métodos mais sofisticados:

- **Baseado em Lógica Fuzzy:** Usa conjuntos fuzzy para detectar bordas, permitindo que cada pixel pertença a múltiplas regiões. Isso oferece uma maior flexibilidade em imagens com transições suaves.
- **Baseado em Algoritmos Genéticos:** Inspirado na teoria da evolução, utiliza processos de seleção, cruzamento e mutação para identificar as bordas de maneira eficiente, especialmente em padrões complexos.
- **Baseado em Redes Neurais:** Redes neurais artificiais são treinadas para aprender padrões de bordas, ajustando os pesos entre suas camadas. São particularmente eficazes em detecção de bordas em cenários com grande variabilidade de padrões.

2.3.2 Métodos Baseados em Regiões (Region-Based Methods)

Os métodos baseados em regiões, por outro lado, focam na similaridade entre os pixels, agrupando-os em regiões contínuas com características semelhantes, como intensidade ou textura.

2.3.2.1 Crescimento de Regiões

O crescimento de regiões é um dos métodos mais intuitivos. Ele começa com um pixel inicial (semente) e expande a região ao adicionar pixels vizinhos que possuem propriedades semelhantes. Esse processo continua até que todos os pixels sejam alocados a uma região.

2.3.2.2 Clustering

- **K-means:** Um dos métodos de clustering mais conhecidos, que particiona a imagem em K grupos ou clusters. Os pixels são atribuídos ao cluster cuja média é mais próxima. É amplamente utilizado, mas pode falhar ao lidar com regiões de forma complexa.
- **Clustering Fuzzy:** Diferente do K-means, esse método permite que um pixel pertença a mais de um cluster, proporcionando mais flexibilidade em imagens com transições suaves entre regiões.

2.3.2.3 Técnica de Divisão e Fusão

Neste método, a imagem é inicialmente dividida em regiões menores e, em seguida, regiões adjacentes são fundidas com base em critérios de similaridade. Essa abordagem é ideal para evitar problemas de supersegmentação.

2.3.2.4 Corte Normalizado

Baseado em teoria dos grafos, o corte normalizado visa minimizar o número de cortes entre regiões, resultando em uma segmentação mais eficiente. Cada pixel é tratado como um vértice no grafo, e as arestas são ponderadas com base na similaridade entre os pixels. Esse método é amplamente utilizado em aplicações de segmentação médica.

2.4 Inteligencia artificial

2.5 Segment Anything

2.6 Trabalhos Correlatos

Nesta seção são apresentados os trabalhos correlatos ao proposto neste trabalho

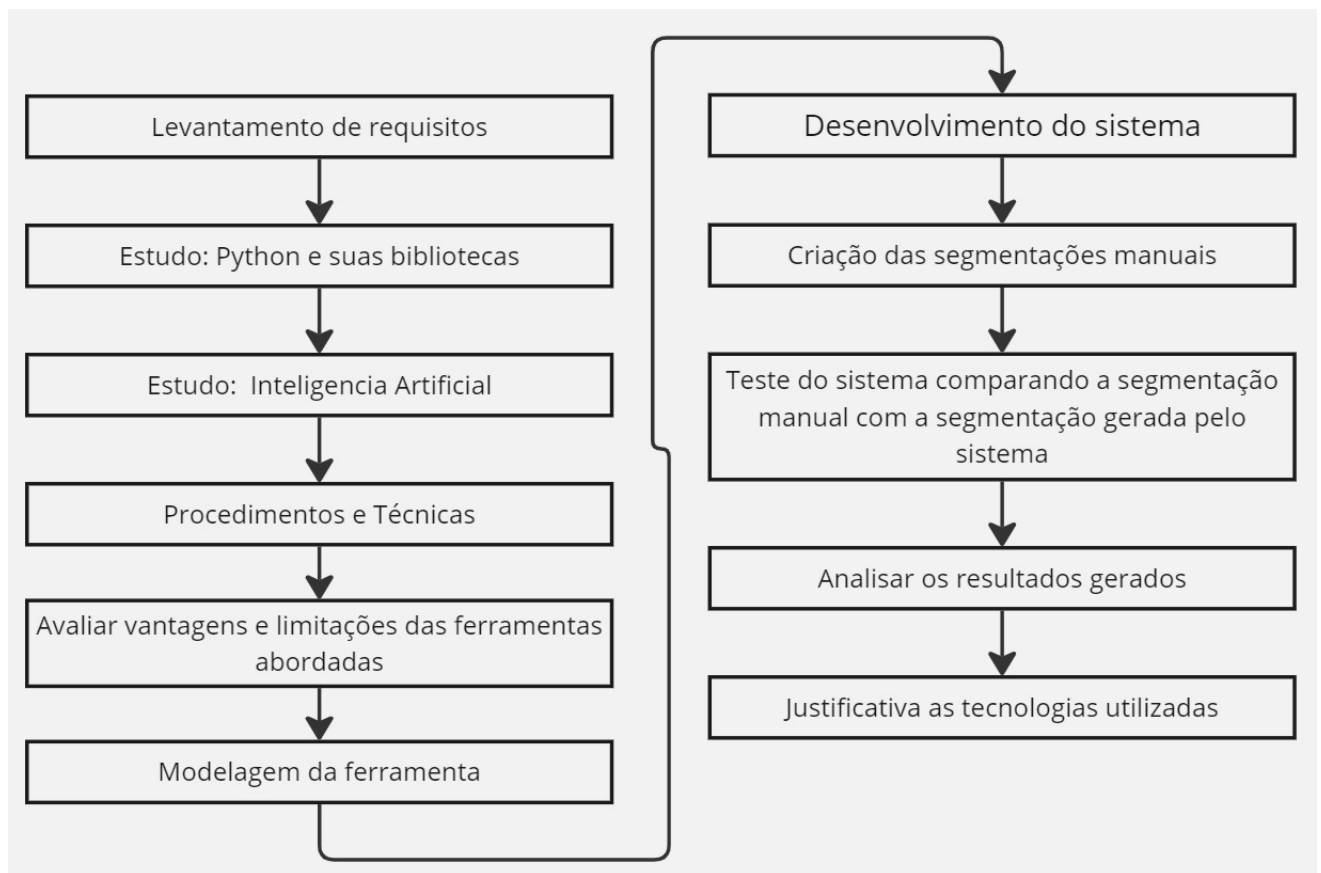
2.6.1 Segmentation by grouping junctions

Escrito por Ishikawa e Geiger (1998) este trabalho nos apresenta uma abordagem inovadora para a segmentação de imagens em tons de cinza, focando em junções, que são pontos críticos onde as bordas se encontram. Este método enfatiza a importância dessas junções na definição de regiões significativas dentro de uma imagem, contrastando com métodos tradicionais que frequentemente dependem apenas de informações de intensidade ou gradiente.

3 METODOLOGIA

Descrever metodologia, materiais e métodos utilizados no estudo, bem como os procedimentos experimentais realizados, nesta etapa será descrito vários assuntos sobre os passos a serem realizados iniciando com o levantamento dos dados, para uma análise e estudo posteriormente analisando todos os processos que são realizados, afim de iniciar o desenvolvimento.

Figura 2 – Passos feito na plataforma miro



Fonte: Feito pelo autor

3.1 Levantamento de Requisitos

Nessa sessão será abordado a pesquisa qualitativa feita a partir da ferramenta *Relight* que usa de um algoritmo para identificação de camadas em uma imagem, assim, quando a imagem é colocada na ferramenta ela faz um mapeamento das áreas altas e baixas para criar as camadas, por esse motivo as camadas que são denominadas como mais alta recebe mais luz do que as mais baixas.

Porém a ferramenta tem uma funcionalidade a mais onde é possível definir em que nível está a luz, fazendo com que a iluminação do objeto possa vir das camadas mais a baixo para as camadas acima, esta ferramenta com a luz funciona como um pincel de adicionar em softwares de pintura, pois ela recebe a luz já existente na foto e complementa por cima com a luz própria, podendo ser de qualquer cor.

A seguir será mostrado na tabela, onde terá a avaliação qualitativa referente a vários tipos de imagens testadas a partir da ferramenta, que para efeitos deste trabalho serão utilizados os valores de alta resolução como acima de 720 pixels, média resolução entre 720 e 256 pixels e baixa resolução abaixo de 256 pixels, retratado na tabela:

Avaliação qualitativa da ferramenta

Imagem	Alta Resolução	Média Resolução	Baixa Resolução
Objetos	Funciona	Funciona	Restrição
Humanos	Funciona	Funciona	Restrição
Paisagens	Funciona	Problema	Problema

Nessa tabela, foi adotado três categorias para descrever o funcionamento: Funciona, Restrição e Problema. A primeira categoria é utilizada para descrever que a imagem utiliza se comportou de forma correta, Restrição é a categorização para as imagens que funcionam mas em alguns casos ou áreas da imagem geram problemas, já a categoria Problema indica os casos onde a luz não reconhecia as formas de maneira correta.

3.2 Estudo: Python e suas bibliotecas

Nesta seção, será abordado sobre as diferentes bibliotecas estudadas para que a análise pudesse ser feita com o melhor aproveitamento da linguagem. será mostrado também alguns tópicos referente ao estudo do python como linguagem em geral.

Para iniciar o estudo das bibliotecas antes, deve se entender melhor sobre os ambientes do python como o ambiente virtual, além de estudar como é o funcionamento das IDEs com o ambiente python dá mesma forma.

3.2.1 PyTorch

Para o uso da SegmentAnything o pyTorch é uma das dependencias cruciais, já que ele aborda todo o processamento usado por placas de video e sistemas integrados com maior facilidade. O estudo dessa biblioteca se faz necessário apenas para a instalação de seus softwares dependentes como o uso do NVIDIA CUDA entre outros sistemas terceiros para o funcionamento do Segment Anything.

3.2.2 Scikit-image

O Scikit será extremamente necessário para a análise dos recursos e resultados obtidos, pois é com ele que conseguimos facilmente gerar as estatísticas como o MSE por exemplo. Não apenas isso mas o estudo dessa ferramenta deve vir análoga ao estudo dos métodos de estimativa comparando duas imagens.

3.2.3 OpenCV

O OpenCV (`cv2`) será empregado para lidar com tarefas relacionadas ao processamento de imagens, como leitura, conversão e redimensionamento. Ele será utilizado para ler as imagens a partir do disco (`cv2.imread`) e convertê-las entre diferentes espaços de cores para garantir a consistência de dados durante o processamento. Por exemplo, a função `cv2.cvtColor` será usada para converter as imagens do formato BGR para RGB, alinhando-as ao formato de cores esperado pelo modelo de segmentação. Além disso, o OpenCV será usado para redimensionar as imagens (`cv2.resize`) a fim de assegurar que elas possuam o mesmo tamanho antes de realizar comparações e cálculos de correlação. Um dos métodos-chave será o cálculo da Correlação Cruzada Normalizada (NCC), que será realizado por meio da função `cv2.matchTemplate`. Esse método permitirá medir a similaridade entre a imagem segmentada e a imagem esperada, fornecendo uma métrica quantitativa para avaliar a precisão do processo de segmentação.

3.2.4 Numpy

No projeto, o `numpy` será fundamental para a manipulação eficiente de arrays multidimensionais, que representarão as imagens processadas. O `numpy` será utilizado para criar e gerenciar esses arrays, facilitando a realização de operações matemáticas e comparações pixel a pixel nas imagens. Por exemplo, a função `np.zeros` será utilizada para inicializar uma matriz de zeros que servirá como base para a imagem segmentada final na função de geração de máscara. Além disso, o `numpy` permitirá a execução de operações eficientes, como a comparação entre arrays de imagens para verificar a igualdade de pixels com a função `np.array_equal`. As imagens serão normalizadas convertendo os valores de pixel para um formato de ponto flutuante, possibilitando uma análise mais precisa ao dividir os valores por 255.0.

3.3 Estudo: Inteligência Artificial

Nesse estudo será demonstrado algumas das partes que serão necessárias para a criação da análise posteriormente, inicialmente foi preciso uma serie de estudos sobre todas as possibilidades relacionadas as possíveis segmentações realizadas, a seguir cada tipo de segmentação será melhor explicada.

3.3.1 Segmentação por máscara

A segmentação por máscara utiliza um modelo para identificar e isolar regiões específicas dentro de uma imagem. O modelo gera uma máscara binária onde a região segmentada é representada por 1 e o fundo por 0. Este método é eficaz para separar objetos ou áreas de interesse em imagens, permitindo uma análise mais detalhada dessas regiões específicas.

3.3.2 Segmentação por pontos de clique

Neste método, o usuário seleciona pontos dentro da área que deseja segmentar. É utilizado esses pontos como referências para definir os limites da região de interesse. A segmentação é então ajustada com base na localização desses pontos, oferecendo um controle mais preciso sobre as regiões segmentadas, especialmente em imagens complexas.

3.3.3 Segmentação por caixa de delimitação

Aqui, o usuário fornece uma caixa delimitadora ao redor do objeto ou área de interesse. O modelo então realiza a segmentação dentro dessa caixa. Esse método é útil para rapidamente identificar e segmentar objetos que podem ser facilmente contidos dentro de uma área retangular, facilitando a definição da região de interesse.

3.3.4 Segmentação por texto

A segmentação por texto permite ao usuário descrever a região ou objeto desejado em termos textuais. O modelo utiliza essa descrição para identificar e segmentar a área correspondente na imagem. Este método é útil quando a descrição do objeto é mais clara em palavras do que em detalhes visuais, oferecendo uma maneira eficiente de segmentar com base em descrições contextuais.

3.3.5 Segmentação interativa

Este método combina várias abordagens interativas, como pontos de clique e caixas de delimitação, permitindo que o usuário refine a segmentação com base no feedback contínuo. A segmentação inicial pode ser ajustada e melhorada conforme o usuário fornece mais informações ou faz ajustes na área segmentada, resultando em uma segmentação mais precisa e adaptada às necessidades específicas.

3.3.6 Segmentação automática

Na segmentação automática, é aplicado algoritmos de segmentação sem intervenção do usuário. O modelo usa padrões aprendidos para identificar e segmentar automaticamente

os objetos de interesse na imagem. Esse método é ideal para processar grandes volumes de dados ou quando a segmentação precisa ser realizada de forma rápida e eficiente em cenários bem definidos.

3.4 Procedimentos e Técnicas

Para o desenvolvimento desta pesquisa, foram realizados testes preliminares explorando toda a capacidade do modelo Segment Anything (SAM), com o intuito de validar a eficácia dos processos envolvidos na análise de segmentação de imagens. Durante essa fase inicial, foi constatado que o tempo de processamento se mostrou elevado, mesmo em um sistema com hardware moderno. Como resultado, optou-se por não utilizar a tecnologia NVIDIA CUDA, uma vez que o uso intensivo de processamento pela GPU não seria imprescindível para o escopo do projeto.

Com essa decisão tomada, iniciou-se uma nova bateria de testes envolvendo diferentes tipos de imagens, abrangendo diversas categorias visuais, a fim de avaliar o potencial de segmentação do SAM em variados contextos. Os resultados foram, em sua maioria, satisfatórios, com o modelo demonstrando desempenho adequado na maioria dos casos analisados.

Posteriormente, iniciou-se o desenvolvimento de imagens específicas, criadas pelo autor em estilo pixel art, para validar a performance do modelo em situações de segmentação por cor. Essa etapa exigiu ajustes no código, especialmente no que diz respeito à manipulação das camadas e formatos das imagens, além da tipagem adequada em Python. Esse processo envolveu pesquisas extensivas e múltiplas tentativas até a obtenção de um resultado satisfatório na geração de grupos de cores a partir das imagens.

Com o sistema de segmentação de imagens em funcionamento, iniciou-se a etapa de criação manual das camadas de segmentação. O autor definiu como deveriam ser os resultados esperados da segmentação proposta pela IA, segmentando um conjunto de 50 imagens em estilo pixel art com base em cores específicas para realizar as análises. A partir desse ponto, foi possível aplicar métodos existentes para a comparação entre duas imagens.

Inicialmente, foi desenvolvido um método próprio, denominado pelo autor como "Método de Espalhamento". Esse método parte do canto da imagem e realiza uma varredura pixel a pixel ao redor de cada ponto analisado, comparando diretamente com o resultado esperado. Caso a cor do pixel na imagem gerada seja diferente da cor prevista no resultado esperado, era gerada uma pontuação negativa. Se as cores coincidisse, o método prosseguia, continuando a busca por outras colorações na imagem base.

Sempre que vemos uma cor clara e logo depois uma escura, isso significa que há um relevo muito grande ali, por exemplo na Figura 3

Figura 3 – História em quadrinhos, Batman



Fonte: (Bazela, 2022)

Nesse quadrinho do Batman, podemos ver que em seu ombro está muito claro e logo acima onde sua capa está vem uma cor muito escura, isso nos mostra que a profundidade da capa é grande, ao ponto de não chegar nenhuma luz até ela, é claro que precisamos levar em consideração que nas HQ's em geral os contrastes são muito maiores, porque traz esse volume nos trajes.

Partindo para o ramo da *Unity* será preciso um aprendizado todo relacionado a iluminação dentro da ferramenta, uso dos objetos em cena e manipulação deles, pois para o desenvolvimento será necessário a criação de uma malha que forme a imagem, um sistema de camadas para que possa estabelecer profundidade, um sistema de alteração e manipulação da posição, cor e luminosidade do ponto de luz. E por fim um estudo básico de toda interface da *Unity*.

Outro ponto importante no estudo, será a linguagem de programação **C#** que é usada como alicerce para qualquer código que precise ser estruturado lá dentro, desde instanciar objetos dentro da cena até modificar configurações de câmera como movimento, posicionamento e angulo até para recebimento dos arquivos

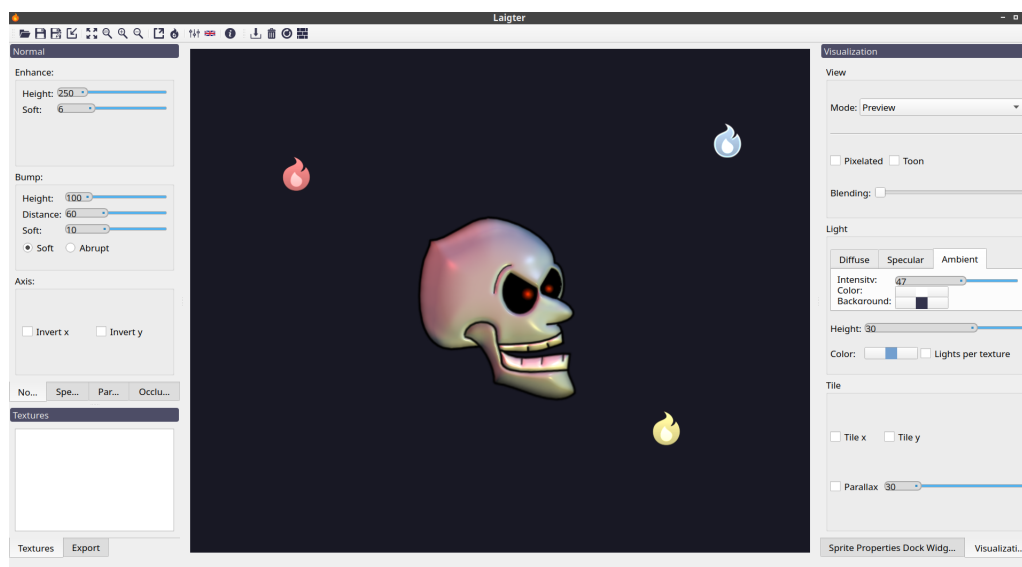
3.5 Avaliar vantagens e limitações das ferramentas abordadas

O *Relight* é uma ferramenta muito boa para processos de iluminação tanto para cenários como personagens, com uma boa modificação como profundidade, posição, cor e luminosidade, dando para o usuário a liberdade de dar personalidade as suas imagens.

Mas ainda existe várias limitações e elas que serão retratadas nessa seção, como foi visto na análise a cima na tabela é possível notar que quanto menor a imagem é, menos funcional ela passa a ser, por exemplo imagens em baixa resolução ou pixeladas começam a ter muitos problemas pois a luz não consegue distinguir os objetos na cena e muito menos paisagens pois com tantas cores perto uma das outras transforma a cena em uma desordem visual.

Além da ferramenta abordada acima existe também algumas outras opções no mercado para este tipo de processo, como o *laigter* mas que ainda se envolve em uma limitação, que é o mapa de normais em apenas uma camada, o *laigter* atual na imagem fornecida pelo usuário como retratado na 4.

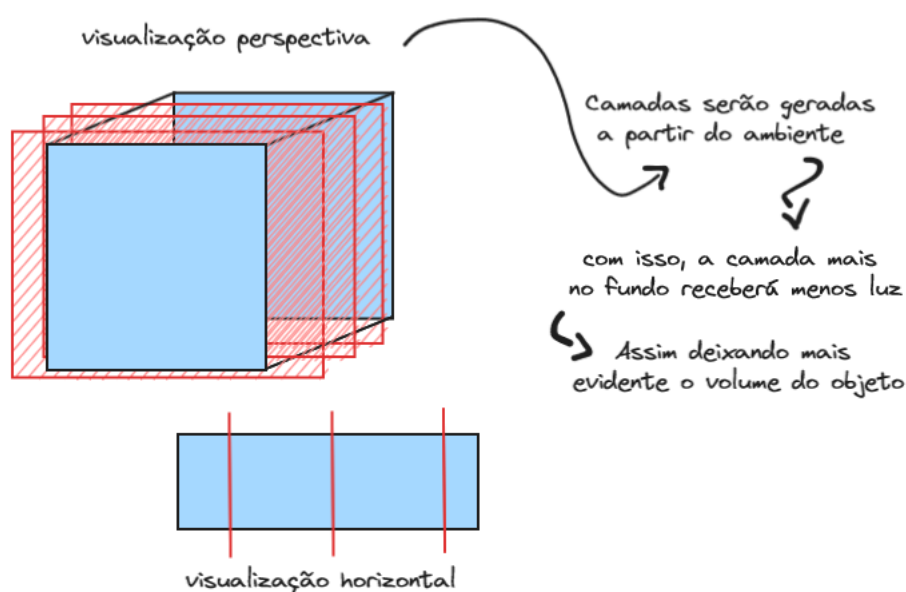
Figura 4 – Imagens retiradas do github da ferramenta



3.6 Modelagem da ferramenta

Uma malha deverá ser criada por cima de toda a imagem e definir um inteiro para a profundidade, cada *pixel* da imagem deve receber uma profundidade como pode se observar na Figura 5

Figura 5 – Rascunho elaborado pelo autor



Assim ele vai numerar toda a malha de *pixels* na tela com números que podem ter uma variação maior dependendo do tamanho da imagem (no caso de exemplo)

Será calculado de 1 a 9, mas com imagens imensas em alta resolução podemos pensar em usar de 1 a 100 ou até mais

Podemos pensar que dessa forma será possível utilizar essa profundidade para produzir uma luz pelas laterais ou pela frente e até atrás de elementos, sem perder sua funcionalidade até mesmo em imagens pequenas

Além disso, cada um desses valores definidos em cada *pixel* da imagem, vão servir de referencia para criação de vários objetos, ele levará em consideração a distância e a variação de cor que foi atingida

Como por exemplo, se existir um objeto a frente da camada 1 até a 10 e um objeto atrás que está na camada 15, essa distancia de 5 camadas irá fazer o objeto dá frente se separar com o de trás transformando a cena com dois objetos ao invés de um

3.7 Justificativa das Tecnologias a serem adotadas

Para a criação da ferramenta de análise será utilizado o ambiente de desenvolvimento Visual Studio Code utilizando Python, foi optado este ambiente por sua praticidade com os diversos formatos de arquivos que serão utilizados ao decorrer do projeto. O python também servirá como unica linguagem de programação para a ferramenta pela facilidade com a criação e uso das tecnologias de inteligencia Artificial que irá adiantar muitos dos processos necessários para a criação da análise

3.8 Avaliação Qualitativa

3.9 Objetivo da Metodologia

3.10 Design da Pesquisa

3.11 Amostragem

3.12 Procedimentos Analíticos

3.13 Critérios de Validação e Confiabilidade

3.14 Aspectos Éticos

3.15 Cronograma e Recursos

3.16 Limitações

4 ANÁLISE DOS RESULTADOS PRELIMINARES

Inicialmente como resultado foi obtido a inicial análise feita na plataforma de iluminação *Relight* que nos traz diversos pontos importantes para a criação da ferramenta.

Foi diagnosticado diversas funções e possíveis formas de como elas podem ter sido elaboradas. como por exemplo a utilização de camadas da imagem e a somatório de cores para criar a intensidade de luz sobre o que é apresentado.

4.1 Cronograma do Trabalho

Segue abaixo o cronograma de trabalho das atividades realizadas e das que serão executadas até a Avaliação Final de TCC.

1. Elaboração da Introdução
2. Elaboração da Revisão da Literatura
3. Elaboração da Metodologia
4. Estudo sobre a *Unity* e Iluminação
5. Desenvolvimento da ferramenta
6. Elaboração dos Resultados e Conclusões

[illegible]

5 CONCLUSÕES PARCIAIS

Foi descrita a modelagem da ferramenta própria, que envolve a criação de uma malha por cima da imagem para definir a profundidade e a variação de cor, permitindo a manipulação da luz. Além disso, foi citado as tecnologias adotadas, como o Visual Studio para programação em *C#* e a Unity como a game *engine* escolhida.

Para avaliar a eficácia da ferramenta, foi planejado realizar uma avaliação qualitativa dos resultados obtidos, incluindo a análise da alteração de imagens, a viabilidade da ferramenta e a eficácia dos cálculos realizados. Para fim é esperado que a ferramenta esteja concluída mesmo que de forma manual, para que ocasionalmente possa ter um progresso posterior em outros trabalhos.

REFERÊNCIAS

- BAZELA, C. **Batman: 3 quadrinhos que influenciaram o novo filme**. 2022. <<https://www.boletimnerd.com.br/batman-3-quadrinhos-que-influenciaram-o-novo-filme/>>. Accessed: 2023-5-20. 16
- CEZIMBRA, D. J. **História da Arte e Design**. 1. ed. [S.l.]: Contentus, 2021. 142 p. 5
- GRAFITTI. **Aprenda a criar efeitos de luz e sombra para as suas ilustrações**. 2019. Disponível em: <<https://blog.grafittiartes.com.br/aprenda-criar-efeitos-de-luz-e-sombra-para-as-suas-ilustracoes>>. Acesso em: 27 de mar. de 2023. 5
- ISHIKAWA, H.; GEIGER, D. **Segmentation by grouping junctions**. [S.l.]: IEEE Computer Society, 1998. 125-131 p. 10
- SHIRLEY, P.; MORLEY, R. **Realistic Ray Tracing, Second Edition**. Taylor & Francis, 2008. ISBN 9781568814612. Disponível em: <<https://books.google.com.br/books?id=knpN6mnhJ8QC>>. 6
- TANEJA, A.; RANJAN, P.; UJJLAYAN, A. A performance study of image segmentation techniques. In: **2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions)**. [S.l.: s.n.], 2015. p. 1–6. 7
- ZAITOUN, N. M.; AQEL, M. J. Survey on image segmentation techniques. **Procedia Computer Science**, v. 65, p. 797–806, 2015. ISSN 1877-0509. International Conference on Communications, management, and Information technology (ICCMIT'2015). Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1877050915028574>>. 7