

UNIVERSIDADE FEDERAL DE JUIZ DE FORA  
INSTITUTO DE CIÊNCIAS EXATAS  
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

# **Estimador de Sombras em Realidade Aumentada para Sistemas Móveis**

**André Luís Braga Dutra**

JUIZ DE FORA  
MARÇO, 2025

# Estimador de Sombras em Realidade Aumentada para Sistemas Móveis

ANDRÉ LUÍS BRAGA DUTRA

Universidade Federal de Juiz de Fora

Instituto de Ciências Exatas

Departamento de Ciência da Computação

Bacharelado em Ciência da Computação

Orientador: Rodrigo Luis de Souza da Silva

JUIZ DE FORA

MARÇO, 2025

# ESTIMADOR DE SOMBRAS EM REALIDADE AUMENTADA PARA SISTEMAS MÓVEIS

André Luís Braga Dutra

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

Rodrigo Luis de Souza da Silva  
Doutor em Engenharia

Marcelo Caniato Renhe  
Doutor em Engenharia de Sistemas e Computação

Luiz Maurílio da Silva Maciel  
Doutor em Engenharia de Sistemas e Computação

JUIZ DE FORA  
11 DE MARÇO, 2025

*Aos meus pais e irmão.*

## Resumo

Este trabalho propõe uma abordagem geométrica para a estimativa do vetor de luz direcional em cenas de realidade aumentada (RA), utilizando técnicas de segmentação de imagem e renderização inversa. A solução foi projetada para ser aplicada em dispositivos móveis, dispensando hardware especializado e algoritmos baseados em aprendizado de máquina, como *deep learning*. O método utiliza câmeras comuns e marcadores fiduciais para projetar sombras realistas em objetos virtuais, permitindo uma integração visual coerente com o ambiente real. Os experimentos realizados demonstraram que a abordagem proposta obteve um erro angular médio de  $11,42^\circ$ , sendo capaz de estimar a iluminação da cena e gerar projeções de sombras convincentes. Os testes qualitativos indicaram que o sistema funciona bem em diversas condições de iluminação, embora apresente limitações em cenários com superfícies texturizadas ou iluminação difusa. Os resultados sugerem que essa solução pode ser uma alternativa eficiente e acessível para aplicações de RA em dispositivos móveis, proporcionando maior imersão e realismo sem exigir infraestrutura computacional complexa.

**Palavras-chave:** Realidade Aumentada, Estimativa de Iluminação, Reconstrução 3D.

# Abstract

This work proposes a geometric approach for estimating the directional light vector in augmented reality (AR) scenes, utilizing image segmentation techniques and inverse rendering. The solution is designed for mobile devices, eliminating the need for specialized hardware and machine learning-based algorithms such as deep learning. The method employs standard cameras and fiducial markers to project realistic shadows on virtual objects, ensuring coherent visual integration with the real environment. The experiments demonstrated that the proposed approach achieved an average angular error of  $11.42^\circ$ , effectively estimating scene illumination and generating visually convincing shadow projections. Qualitative tests indicate that the system performs well under various lighting conditions, although it faces limitations in scenarios with textured surfaces or diffuse lighting. The results suggest that this solution can serve as an efficient and accessible alternative for AR applications on mobile devices, enhancing immersion and realism without requiring complex computational infrastructure.

**Keywords:** Augmented Reality, Lighting Estimation, 3D Reconstruction.

## Agradecimentos

Aos meus pais, avós e irmão, pelo amor, apoio e torcida durante toda minha trajetória acadêmica. Sem vocês, nada disso seria possível.

Aos meus amigos, pelo apoio e encorajamento. Vocês tornaram essa jornada mais leve e memorável.

Ao professor Rodrigo pela orientação, paciência e amizade nos últimos anos, sem a qual este trabalho não se realizaria.

Aos professores do Departamento de Ciência da Computação pelos seus ensinamentos e aos funcionários do curso, que durante esses anos, contribuíram de algum modo para o nosso enriquecimento pessoal e profissional.

*“Milagres só acontecem com as pessoas  
que não desistem.”*

*Emporio Ivankov, One Piece (Eiichiro  
Oda)*

# Conteúdo

<b>Lista de Figuras</b>	<b>7</b>
<b>Lista de Tabelas</b>	<b>8</b>
<b>Lista de Abreviações</b>	<b>9</b>
<b>1 Introdução</b>	<b>10</b>
1.1 Contextualização . . . . .	10
1.2 Descrição do Problema . . . . .	11
1.3 Justificativa . . . . .	12
1.4 Material e Métodos . . . . .	12
1.5 Objetivos . . . . .	13
<b>2 Fundamentos</b>	<b>15</b>
2.1 Realidade Aumentada . . . . .	15
2.2 Renderização Inversa . . . . .	16
2.3 Segmentação de Imagens . . . . .	17
2.4 Considerações Finais . . . . .	19
<b>3 Trabalhos relacionados</b>	<b>20</b>
3.1 Estimativa de Fontes de Luz com Geometria . . . . .	20
<b>4 Sistema Proposto</b>	<b>25</b>
4.1 Arquitetura do Sistema . . . . .	25
4.2 Funcionamento do Sistema . . . . .	26
4.2.1 Entrada de Dados . . . . .	28
4.2.2 Segmentação de imagens . . . . .	28
4.2.3 Associação entre Objeto e Sombra . . . . .	29
4.2.4 Cálculo do Vetor de Luz Direcional . . . . .	30
4.3 Resumo . . . . .	31
<b>5 Resultados</b>	<b>33</b>
5.1 Cenários Artificiais . . . . .	33
5.1.1 Resultados Quantitativos . . . . .	34
5.2 Cenários Reais . . . . .	36
5.2.1 Resultados qualitativos . . . . .	36
5.3 Limitações do sistema . . . . .	38
5.4 Resumo dos resultados . . . . .	40
<b>6 Conclusão</b>	<b>42</b>
<b>Bibliografia</b>	<b>44</b>

## Lista de Figuras

4.1	Diagrama do <i>pipeline</i> de processamento do sistema. . . . .	27
4.2	Visualização do <i>pipeline</i> do sistema. . . . .	27
5.1	Comparação entre <i>Ground Truth</i> e Resultados Estimados em todos os cenários artificiais. . . . .	34
5.2	Resultados em cenários reais com cubo texturizado virtual. . . . .	36
5.3	Visualização das cenas com diferentes objetos 3D. . . . .	37
5.4	Erros do sistema. . . . .	38

## Lista de Tabelas

3.1	Comparação entre trabalhos relacionados . . . . .	23
5.1	Erro angular entre os vetores reais e os vetores estimados. . . . .	35

## Lista de Abreviações

CNN	<i>Convolutional Neural Network</i>
HMD	<i>Head Mounted Display</i>
RGB	<i>Red, Green, Blue</i>
RGB-D	<i>Red, Green, Blue - Depth</i>
HDR	<i>High Dynamic Range</i>
LDR	<i>Low Dynamic Range</i>
RA	Realidade Aumentada
VR	<i>Virtual Reality</i>
SH	<i>Spherical Harmonics</i>
IRT	<i>Inverse Ray Tracing</i>
IPT	<i>Inverse Path Tracing</i>
DR	<i>Differentiable Rendering</i>
GMM	<i>Gaussian Mixture Models</i>

# 1 Introdução

A realidade aumentada (RA) é descrita como uma tecnologia que integra elementos virtuais e reais de maneira interativa e em tempo real (AZUMA; RONALD, 1997). Essa tecnologia tem se expandido para diversas áreas, como jogos, educação, indústria e saúde, oferecendo novas formas de interação e visualização de informações. Para que as cenas em RA sejam convincentes, é essencial que os objetos virtuais se integrem ao ambiente real, garantindo sombras, geometria e iluminação consistentes.

O presente estudo concentra-se no cálculo do vetor de luz direcional em cenas de RA, possibilitando a projeção de sombras precisas para objetos virtuais. O cálculo é realizado por meio de técnicas de processamento de imagem e renderização inversa, que estimam a direção da luz na cena. Em vez de recorrer a abordagens baseadas em *deep learning*, são utilizados métodos geométricos, que oferecem uma solução mais simples e direta. Essa abordagem permite a projeção de sombras coerentes com o ambiente, aumentando a imersão e o realismo nos cenários de realidade aumentada.

Os dispositivos mais comuns que utilizam RA incluem *Head-Mounted Displays* (HMD), como óculos de realidade aumentada, e *smartphones*, que se destacam por sua versatilidade e acessibilidade. Embora diversos dispositivos suportem essa tecnologia, o foco deste trabalho está na aplicação de RA em celulares, dada a sua popularidade e a capacidade de proporcionar uma experiência interativa.

## 1.1 Contextualização

Com o avanço dos dispositivos móveis nas últimas décadas, a realidade aumentada (RA) tornou-se mais acessível e presente em várias áreas. Apesar disso, a integração visual adequada entre objetos virtuais e o ambiente real, especialmente no que diz respeito à iluminação e projeção de sombras, continua sendo um grande desafio, como destacado por Cao e Foroosh (2007), que abordam a complexidade de estimar a direção da luz em cenas de RA utilizando sombras solares. Uma iluminação inconsistente compromete o realismo

e imersão das cenas de RA.

Estudos anteriores abordaram a estimativa de luz em RA com base em métodos geométricos e técnicas mais avançadas. Cao e Foroosh (2007) e Koc e Balcisoy (2013), por exemplo, calcularam a direção da luz com base em sombras e geometria de objetos, enquanto métodos baseados em *deep learning*, como o de LeGendre et al. (2019), oferecem maior flexibilidade para cenários complexos. No entanto, abordagens baseadas em *deep learning* exigem grandes volumes de dados para treinamento e podem demandar hardware especializado, o que limita sua aplicação em dispositivos móveis.

O presente estudo propõe uma abordagem geométrica que não depende de hardware especializado, utilizando apenas uma câmera comum e marcadores fiduciais para calcular o vetor de luz direcional. Essa abordagem otimiza o cálculo de sombras e iluminação para dispositivos móveis, evitando o alto custo computacional e a necessidade de grandes volumes de dados.

## 1.2 Descrição do Problema

A correta projeção de sombras em cenas de realidade aumentada (RA) é essencial para garantir uma imersão convincente e a integração visual entre objetos virtuais e o ambiente real. O problema central está em calcular o vetor de luz direcional de forma precisa, de modo que a iluminação dos objetos virtuais corresponda às condições de iluminação do mundo físico. No entanto, realizar esse cálculo de maneira eficiente, utilizando apenas uma câmera comum e marcadores fiduciais, sem recorrer a hardware especializado ou técnicas complexas, continua sendo um obstáculo significativo.

Esse problema é particularmente importante em cenários de RA aplicados a dispositivos móveis, onde as limitações de processamento e recursos tornam ainda mais desafiador garantir uma projeção de sombras realista e coerente com o ambiente. Como destacado por Kán e Kafumann (2019), a estimativa de iluminação em dispositivos móveis deve ser precisa e eficiente para que os objetos virtuais se comportem visualmente de maneira natural, sem parecerem deslocados no ambiente. Além disso, a variação nas condições de iluminação, como luz direta ou difusa, adiciona uma camada extra de complexidade ao cálculo preciso do vetor de luz, exigindo uma solução robusta que funcione em diferentes

cenários.

## 1.3 Justificativa

A escolha por uma abordagem geométrica e a utilização de câmeras simples de celulares e marcadores fiduciais, em detrimento de técnicas mais complexas como o *deep learning* ou o uso de hardware especializado, se justifica pela necessidade de desenvolver uma solução acessível, eficiente e adequada a dispositivos móveis. Em cenários de realidade aumentada (RA), onde a limitação de processamento e recursos é significativa, métodos que demandam alto poder computacional ou grandes volumes de dados para treinamento podem não ser viáveis.

Técnicas de *deep learning*, como as propostas por LeGendre et al. (2019), embora eficazes em cenários mais complexos, apresentam desafios práticos, como a exigência de hardware especializado e o alto custo computacional, o que pode limitar sua aplicação em dispositivos móveis. A abordagem deste trabalho, por outro lado, se baseia em métodos geométricos que utilizam informações diretamente extraídas da cena por meio de uma única câmera e marcadores fiduciais, como os oferecidos pelo *ArToolKit* (KATO; BILLINGHURST, 1999). Isso possibilita uma solução de baixo custo, aplicável em contextos onde o uso de tecnologias avançadas é inviável.

Além disso, a abordagem proposta visa manter a eficiência, sem sacrificar a qualidade das estimativas de iluminação e sombras, essenciais para uma boa integração entre os objetos virtuais e o ambiente real. Essa escolha é particularmente relevante em aplicações voltadas para dispositivos móveis, onde a demanda por tempo real e baixo consumo de recursos é crítica. A justificativa para o desenvolvimento dessa solução está na busca por uma ferramenta acessível, eficiente e prática para projetar sombras precisas em cenas de RA, sem a necessidade de grandes investimentos em infraestrutura tecnológica.

## 1.4 Material e Métodos

O método proposto neste trabalho para estimar o vetor de luz direcional em cenas de realidade aumentada (RA) segue um *pipeline* de etapas interligadas, utilizando recursos

simples e acessíveis, como câmeras de celulares e marcadores fiduciais. As câmeras de dispositivos móveis permitem a captura de cenas em tempo real, sem a necessidade de equipamentos especializados, como câmeras de profundidade ou múltiplas câmeras.

Os marcadores fiduciais são amplamente utilizados em sistemas de RA para determinar a posição e a orientação dos objetos virtuais em relação ao ambiente real. Esses marcadores, como os oferecidos pela ferramenta *ArToolKit* desenvolvida por Kato e Billinghurst (1999), servem como pontos de referência na imagem capturada, facilitando a correlação entre o mundo real e o virtual. Sua aplicação simplifica a projeção de objetos virtuais e, neste contexto, contribui diretamente para o cálculo do vetor de luz direcional.

O *pipeline* proposto envolve, primeiramente, a captura de uma imagem 2D da cena usando a câmera de um celular. Em seguida, os objetos e sombras presentes na imagem são separados dos demais elementos utilizando técnicas de processamento de imagem. Após essa separação, o centro de massa dos objetos e suas respectivas sombras são calculados. Com essas informações, os dados da imagem 2D são transpostos para o ambiente tridimensional por meio de renderização inversa, permitindo o cálculo do vetor de luz direcional.

Esse processo, que utiliza recursos limitados como uma única câmera e marcadores fiduciais, proporciona uma solução eficiente e de baixo custo para estimativa de luz em dispositivos móveis, dispensando o uso de hardware especializado ou algoritmos complexos de *deep learning*.

## 1.5 Objetivos

O presente trabalho tem como objetivo desenvolver uma metodologia eficiente para a estimativa do vetor de luz direcional em cenas de realidade aumentada (RA), utilizando apenas uma câmera de dispositivo móvel e marcadores fiduciais, de modo a garantir a projeção precisa de sombras em objetos virtuais, sem a necessidade de hardware especializado ou algoritmos complexos, como os baseados em *deep learning*.

Especificamente, busca-se implementar um *pipeline* de processamento de imagem capaz de separar objetos e sombras a partir de imagens 2D capturadas por câmeras simples. A partir dessas informações, são aplicadas técnicas de renderização inversa para

---

transpor os dados da imagem 2D para o ambiente tridimensional, possibilitando o cálculo do vetor de luz. O trabalho também visa validar a metodologia em diferentes cenários de RA, avaliando a eficiência e precisão do sistema.

Por fim, é realizada uma comparação dos resultados obtidos com outras abordagens existentes, destacando as vantagens da solução proposta em termos de custo computacional, simplicidade de implementação e aplicabilidade em dispositivos móveis.

## 2 Fundamentos

Neste capítulo são apresentados os conceitos principais que sustentam o desenvolvimento deste trabalho. Primeiramente, discute-se a realidade aumentada (RA), abordando suas características, evolução ao longo dos anos e algumas de suas aplicações mais comuns. Em seguida, é explorada a técnica de renderização inversa, com destaque para métodos como o *Differentiable Rendering* (DR) e o *Inverse Ray Tracing* (IRT), utilizados tanto em trabalhos relacionados quanto no presente estudo.

Por fim, é tratado o processamento de imagens, com foco em métodos de segmentação como o *K-means clustering* e o *GrabCut*, além de técnicas que auxiliam na manipulação das informações visuais, sendo fundamentais para o sucesso das aplicações desenvolvidas ao longo do trabalho.

### 2.1 Realidade Aumentada

A realidade aumentada (RA) é uma tecnologia que combina elementos do mundo real e do mundo virtual, proporcionando uma experiência interativa e imersiva. As primeiras ideias sobre RA começaram a ser desenvolvidas em 1968, com o trabalho pioneiro Sutherland (1968). Esse trabalho desenvolveu o primeiro *Head-Mounted Display* (HMD), conhecido como *Sword of Damocles*, que permitia a sobreposição de gráficos gerados por computador sobre o mundo real. Esse trabalho marcou o início das interfaces que combinam o real com o virtual.

Uma definição mais clara e moderna da RA foi apresentada em Azuma e Ronald (1997). O trabalho destacou três características principais, que envolvem a combinação de objetos reais e virtuais no mesmo ambiente, a interação com esses objetos em tempo real, e o alinhamento correto dos objetos virtuais em um espaço tridimensional. Essas características são essenciais para que os sistemas de RA funcionem de maneira convincente, fazendo com que os objetos virtuais se integrem naturalmente ao ambiente físico e proporcionem uma experiência mais coerente ao usuário.

Com o avanço da tecnologia, a RA se tornou mais acessível, principalmente graças ao desenvolvimento de ferramentas como o *ARToolKit*, descrito em Kato e Billinghurst (1999). O *ARToolKit* introduziu o uso de marcadores fiduciais, permitindo a sobreposição precisa de objetos virtuais em cenas reais. Esses marcadores ajudaram a rastrear com eficiência a posição e a orientação no espaço 3D, algo fundamental para que os objetos virtuais fossem corretamente integrados em sistemas de RA. Isso facilitou a aplicação da RA em dispositivos móveis, tornando a tecnologia acessível a um público muito maior.

Outro aspecto importante da realidade aumentada é a percepção de iluminação e sombras, que tem um papel crucial no realismo visual e na imersão do usuário. A projeção de sombras realistas é essencial para que os objetos virtuais pareçam realmente integrados no ambiente físico, ajudando a fornecer pistas visuais sobre profundidade e posição. Kán e Kafumann (2019) destacam que, em dispositivos móveis, calcular sombras em tempo real, levando em conta as condições de luz natural, faz com que os objetos virtuais se comportem visualmente de maneira semelhante aos objetos reais.

## 2.2 Renderização Inversa

Renderização inversa pode ser descrita como “o problema de estimar uma ou mais propriedades de iluminação, refletância e forma a partir da aparência observada (ou seja, uma ou mais imagens)” (YU; SMITH, 2019). Trata-se de uma técnica que inverte o processo tradicional de renderização, utilizando imagens 2D para deduzir as propriedades físicas de uma cena 3D. Essa abordagem é particularmente relevante em áreas como a RA, onde a integração de objetos virtuais em ambientes reais depende de uma estimativa precisa das condições de iluminação e das características dos materiais.

Nos últimos anos, redes neurais passaram a ser utilizadas na renderização inversa para estimar informações tridimensionais a partir de imagens bidimensionais (KATO et al., 2020). No entanto, o treinamento dessas redes exige grandes volumes de dados 3D, que são mais difíceis de obter em comparação com imagens 2D. Para contornar essa limitação, algumas abordagens recentes utilizam dados 2D como referência, ajustando o treinamento para melhorar a estimativa 3D. Uma das estratégias para isso é integrar o processo de renderização inversa ao *pipeline* da rede neural, permitindo que os resulta-

dos gerados sejam comparados diretamente com imagens reais e refinados com base na diferença observada.

O *Differentiable Rendering* (DR), conforme descrito por Kato et al. (2020), envolve uma série de técnicas que buscam otimizar o processo de renderização, permitindo que o sistema obtenha gradientes úteis do processo de renderização. Isso possibilita que a rede neural ajuste de forma eficiente as entidades 3D, mesmo trabalhando com entradas 2D. Ao integrar o processo de renderização ao treinamento da rede, é possível obter estimativas mais precisas e rápidas de geometria e iluminação da cena.

Azinovic et al. (2019), afirmam que, quando a geometria 3D da cena já é conhecida, outras técnicas, como o *Inverse Ray Tracing* (IRT) ou o *Inverse Path Tracing* (IPT), podem ser mais adequadas para obter informações sobre a iluminação da cena. Essas técnicas permitem uma melhor estimativa da interação da luz com a geometria já definida da cena, especialmente em ambientes de realidade aumentada.

O *Inverse Ray Tracing* (IRT), em particular, é uma técnica que rastreia os raios de luz da cena de volta ao ponto de observação. No contexto de cenas em realidade aumentada, o IRT pode ser utilizado para calcular a distância entre pontos 3D e a câmera, rastreando sombras observadas em imagens 2D. Quando a sombra de um objeto é capturada em uma imagem 2D, o IRT permite mapear essa sombra de volta ao espaço tridimensional, utilizando a posição do centro da sombra para realizar o *ray casting* e determinar sua projeção no plano 3D. Dessa forma, garante-se que as sombras virtuais sejam coerentes com a iluminação real da cena, criando uma integração visual mais natural e precisa entre objetos virtuais e o ambiente físico.

## 2.3 Segmentação de Imagens

A segmentação de imagens consiste em dividir uma imagem em regiões homogêneas, com o objetivo de isolar partes de interesse, tais como sombras e objetos, permitindo a extração de informações relevantes para análise e processamento. Para tal tarefa, este trabalho fundamenta-se em três técnicas principais: o algoritmo *K-means*, o método *GrabCut* e a técnica de *thresholding*.

O *K-means clustering* apresentado em Macqueen (1967), é um algoritmo de agru-

pamento que particiona os dados em  $k$  grupos ou *clusters*. No contexto da segmentação, cada pixel da imagem é representado por suas características, como valores de cor ou intensidade, e é associado ao *cluster* cujo centroide, calculado com base em uma métrica de distância, geralmente a euclidiana, seja o mais próximo. O processo inicia com a seleção aleatória dos  $k$  centroides e, em seguida, os *pixels* são atribuídos aos *clusters* correspondentes. Após essa etapa, os centroides são recalculados como a média dos *pixels* de cada *cluster*, e essa iteração se repete até que as atribuições se estabilizem. Uma particularidade importante do *K-means* é a sensibilidade à escolha inicial dos centroides, o que pode levar a resultados variados para a mesma imagem; para mitigar esse efeito, é comum utilizar uma semente fixa na inicialização. Neste trabalho, a segmentação é realizada no espaço de cor LAB, explorando-se o componente  $L^*$ , que varia de 0 (preto absoluto) a 100 (branco absoluto), para identificar regiões com baixa luminosidade, tipicamente associadas a sombras.

Outro método amplamente utilizado é o *GrabCut*, desenvolvido por Rother, Kolmogorov e Blake (2004). É uma técnica iterativa de segmentação que expande as abordagens de cortes em grafos para a extração de objetos em uma imagem. Inicialmente, é necessário definir um retângulo que delimite a área onde se presume estar o objeto de interesse; esse retângulo serve como ponto de partida para que o algoritmo possa identificar, com base na similaridade em termos de cor e textura, quais *pixels* pertencem ao objeto e quais pertencem ao fundo. A partir dessa definição, o algoritmo constrói um grafo onde cada pixel é representado por um nó, e as conexões entre os *pixels* são ponderadas de acordo com a similaridade em termos de cor e textura. Para modelar as distribuições de cor do objeto e do fundo, utiliza-se um modelo de Mistura Gaussiana (GMM), caracterizado por parâmetros estatísticos como a média ( $\mu$ ) e o desvio padrão ( $\sigma$ ). O *GrabCut* minimiza uma função de energia que combina um termo de dados (avaliando a probabilidade dos *pixels* se ajustarem aos modelos de cor) e um termo de suavidade (impondo a consistência entre *pixels* vizinhos). Essa minimização, realizada por meio de algoritmos de fluxo máximo/mínimo, resulta em uma máscara binária que separa o objeto do fundo. O processo é iterativo, com a atualização contínua dos GMMs e a reavaliação da segmentação, refinando progressivamente os contornos do objeto.

A técnica de *thresholding* é uma abordagem direta para segmentação, utilizada quando há uma distinção clara entre os níveis de intensidade dos objetos e do fundo. Nesse método, um valor de limiar é definido, podendo ser determinado automaticamente, como no método de Otsu et al. (1975), de modo que os *pixels* com intensidade superior ao limiar sejam atribuídos a uma classe (por exemplo, branco, com valor 255) e os que estiverem abaixo, a outra classe (por exemplo, preto, com valor 0). O objetivo é maximizar a separação entre as classes, minimizando a variância intra-classe ou, equivalentemente, maximizando a variância inter-classe. Essa conversão gera uma imagem binária que facilita a criação de máscaras para isolar regiões de interesse, embora sua eficácia dependa da uniformidade da iluminação e da clara distinção entre o objeto e o fundo.

## 2.4 Considerações Finais

Neste capítulo, foi possível explorar diversos conceitos que são essenciais para o desenvolvimento deste trabalho. A realidade aumentada (RA), com sua capacidade de mesclar o mundo real com elementos virtuais, sendo abordada tanto do ponto de vista histórico quanto tecnológico. A evolução da RA mostra como ela tem sido aprimorada ao longo dos anos, permitindo a criação de experiências imersivas e interativas.

Também foram apresentados os métodos de renderização inversa, como o *Differentiable Rendering* e o *Inverse Ray Tracing*, que auxiliam na estimação de características importantes da cena, como sombras e iluminação.

Por último, o capítulo trouxe uma visão geral sobre alguns algoritmos de processamento de imagens utilizados neste trabalho, focando em ferramentas que facilitam a manipulação e análise de dados visuais. Técnicas como o *K-means clustering* e o *GrabCut* desempenham um papel importante ao permitir a segmentação de objetos e a criação de máscaras, que são úteis para várias aplicações práticas deste projeto.

## 3 Trabalhos relacionados

Os métodos geométricos têm sido amplamente utilizados para estimar a iluminação ambiente, analisando a interação da luz com os elementos da cena para inferir sua direção e intensidade (KOC; BALCISOY, 2013). Historicamente, as pesquisas nesse campo começaram utilizando principalmente técnicas de processamento de imagem e métodos geométricos, que analisam a forma dos objetos e a influência da iluminação no ambiente. Esses métodos se destacam por sua simplicidade e eficiência, mas enfrentam limitações em cenários dinâmicos ou com múltiplas fontes de luz.

Nos últimos anos, com o avanço do *deep learning*, surgiram técnicas que têm se mostrado mais robustas na estimação da iluminação em ambientes complexos. No entanto, essas abordagens ainda apresentam desafios, como a necessidade de grandes volumes de dados para treinamento e a dificuldade em lidar com condições de iluminação variadas, o que torna a estimação da luz um problema mal posto, ou seja, sem uma solução única para cada cena (MARQUES et al., 2022).

Dado que a proposta deste trabalho não envolve técnicas de *deep learning*, este capítulo abordará exclusivamente trabalhos que utilizam métodos geométricos para a estimação da iluminação. Serão apresentados estudos que exploram a relação entre luz e sombras na cena, empregando técnicas de processamento de imagem e análise geométrica para inferir a direção da iluminação.

### 3.1 Estimativa de Fontes de Luz com Geometria

A estimativa de fontes de luz com base em geometria é muito utilizada em aplicações de realidade aumentada, efeitos visuais e visão computacional. Esses métodos se baseiam na análise da geometria dos objetos presentes na cena e nas sombras que eles projetam para calcular a direção e intensidade da luz incidente. Uma vantagem desse tipo de abordagem é que ela depende de propriedades físicas simples, como a forma dos objetos e a correspondência das sombras, dispensando o uso de modelos complexos ou grandes volumes de

dados para treinamento. Neste tópico, serão analisados trabalhos que utilizam técnicas geométricas para a estimativa de luz, com foco em ambientes de realidade aumentada.

A técnica desenvolvida em Wang e Samaras (2003) utiliza múltiplas fontes de luz direcionais a partir de uma única imagem para estimar a iluminação de objetos com geometria conhecida e refletância lambertiana. Eles combinaram informações de sombreamento e sombras projetadas para determinar a iluminação em cenas de realidade aumentada, sem a necessidade de objetos de calibração. Comparado a outros métodos, este apresentou melhores resultados ao operar com múltiplas fontes de luz. No entanto, a técnica requer uma geometria conhecida, o que pode limitar sua aplicabilidade em alguns cenários.

A calibração de câmeras e a estimativa da orientação de fontes de luz utilizando sombras solares foi o foco do trabalho de Cao e Foroosh (2007). Nesse trabalho, foi desenvolvido um método para calcular parâmetros como distância focal, razão de aspecto e ponto principal da câmera. O método foi capaz de estimar a direção da luz solar a partir de dois pontos de vista de uma cena, demonstrando precisão tanto em imagens sintéticas quanto reais. O método se destacou por dispensar objetos de calibração complexos, sendo aplicado em ambientes naturais. Contudo, a principal limitação está na dependência da luz solar, o que impossibilita sua aplicação em cenários internos ou com fontes de luz artificiais.

No campo de estimativa de luz a partir de uma única imagem, Nguyen e Le (2012) propuseram um método baseado na utilização de objetos com forma convexa como sondas de luz. O sistema calcula direções e intensidades de luz com base no contorno definido pelo usuário, eliminando a necessidade de sondas físicas de luz e conhecimento prévio da geometria 3D da cena. Esse método demonstrou boa precisão na estimativa de ângulos zenitais, mas encontrou dificuldades ao lidar com superfícies refletivas ou transparentes.

A estimativa de iluminação em tempo real também foi explorada por Gruber, Richter-Trummer e Schmalstieg (2012), que desenvolveram um sistema baseado na geometria arbitrária capturada por câmeras RGB-D. O método utiliza harmônicos esféricos para calcular a iluminação ambiente e renderizar sombras suaves em objetos virtuais, garantindo uma integração visual coerente. O sistema mostrou-se eficiente em lidar com

mudanças dinâmicas nas fontes de luz e na cena, mas sua aplicação é limitada a superfícies lambertianas difusas, o que pode restringir seu uso em materiais reflexivos.

Em cenas de realidade aumentada ao ar livre, a estimativa da iluminação ambiente pode ser feita através de superfícies lambertianas, como foi proposto em Koc e Balcisoy (2013). Neste trabalho, foi utilizada a geometria de rostos humanos para extrair a direção e intensidade da luz ambiente, iluminando objetos virtuais de forma realista. Este sistema demonstrou eficácia em dispositivos móveis, especialmente sob luz solar direta, mas apresentou limitações em cenários nublados, onde a luz difusa impacta a precisão das estimativas. Outro ponto positivo foi a integração com sensores comuns em dispositivos móveis, que permitiu ao sistema operar em tempo real.

O uso de otimização e remoção de albedo para estimar múltiplas fontes de luz foi a proposta de Lopez-Moreno et al. (2013), que criaram um sistema capaz de evitar a fusão de luzes próximas. O método foi aplicado com sucesso em cenários complexos, demonstrando precisão significativa na estimativa de direções e intensidades de luz, com um erro médio de 20-30 graus. Contudo, a técnica depende da seleção manual de objetos convexos, o que limita sua aplicabilidade em cenários com formas geométricas irregulares.

A correspondência entre as bordas das sombras e as superfícies que as projetam é uma técnica que dispensa o conhecimento prévio da geometria do objeto, como mostrado por Chotikakamthorn (2015). O método faz uso de imagens de profundidade RGB-D para estimar a localização de fontes de luz pontuais próximas. Os resultados indicaram que o método é eficaz na estimativa da direção da luz em ambientes internos, mas apresentou maior erro de distância em cenas com objetos menores.

A utilização de câmeras RGB-D, como a do Kinect, também foi explorada por Boom et al. (2017), onde foi demonstrado um método híbrido CPU-GPU para estimar a posição de uma fonte de luz pontual de forma eficiente. O sistema se baseia em mapas de profundidade e imagens de intensidade para realizar cálculos de iluminação, permitindo a renderização realista de objetos sintéticos em cenários de realidade aumentada. Com uma precisão média de 20 graus, o sistema mostrou-se eficaz, embora apresente limitações quando há múltiplas fontes de luz na cena.

Outro método para estimar a direção da luz em cenas de realidade aumentada,

utilizando sombras e objetos em primeiro plano a partir de uma única imagem, foi proposto em Liu e Wu (2022). A técnica se diferencia por empregar uma transformação de homografia para alinhar o sistema de coordenadas da imagem com o do mundo real, aumentando a precisão da estimativa de iluminação. Testes realizados em cenas virtuais demonstraram erros médios baixos, dependendo da cena, com bons resultados em cenas que apresentavam sombras claras. O sistema também integra interação manual direta, permitindo que os usuários toquem em objetos virtuais de maneira mais natural. Contudo, a precisão diminui em cenas onde as sombras estão parcialmente obstruídas ou em casos onde o objeto não está completamente conectado ao chão, e a abordagem é limitada a uma única fonte de luz direcional.

Tabela 3.1: Comparação entre trabalhos relacionados

<b>Trabalho</b>	<b>Entradas</b>	<b>Saídas</b>	<b>Dispositivos Alvo</b>
(WANG; SAMARAS, 2003)	Imagem 2D	Múltiplos Vetores 3D de luz	Câmeras comuns
(CAO; Foroosh, 2007)	Duas Imagens 2D	Vetor 3D da luz	Câmeras comuns
(NGUYEN; LE, 2012)	Imagem 2D	Múltiplos Vetores 3D de luz	Câmeras comuns
(GRUBER et al., 2012)	Imagens RGB-D	Iluminação Harmônicas Esféricas	Câmeras RGB-D
(KOC; BALCISOY, 2013)	Imagem 2D	Vetor 3D da luz	<i>Mobile</i>
(LOPEZ et al., 2013)	Imagem 2D	Múltiplos Vetores 3D de luz	Câmeras comuns
(BOOM et al., 2015)	Imagens RGB-D	Vetor 3D da luz	Câmera RGB-D
(CHOTIKA-KAMTHORN, 2015)	Imagens RGB-D	Múltiplos Vetores 3D de luz	Câmera RGB-D
(LIU; WU, 2022)	Imagem 2D	Vetor 3D da luz	Câmeras comuns e <i>Mobile</i>

A Tabela 3.1 permite visualizar as diferentes abordagens para obter a iluminação da cena. Os métodos geométricos para a estimativa da iluminação exploram diferentes estratégias para recuperar informações sobre a luz a partir de imagens. Alguns métodos utilizam sensores de profundidade, como câmeras RGB-D, para reconstruir a posição

tridimensional da fonte luminosa, aproveitando informações de profundidade e cor simultaneamente (BOOM et al., 2017; CHOTIKAKAMTHORN, 2015; GRUBER; RICHTER-TRUMMER; SCHMALSTIEG, 2012). Outras abordagens operam exclusivamente com imagens 2D, analisando a relação entre sombras projetadas e os objetos na cena para estimar a direção da iluminação (CAO; FOROOSH, 2007; KOC; BALCISOY, 2013; LIU; WU, 2022). Além disso, algumas técnicas utilizam modelos geométricos predefinidos, como superfícies lambertianas ou objetos convexos, para inferir múltiplas direções de luz (WANG; SAMARAS, 2003; NGUYEN; LE, 2012; LOPEZ-MORENO et al., 2013). Cada uma dessas abordagens apresenta vantagens e desafios distintos, sendo mais adequadas a diferentes tipos de cena e aplicações, desde ambientes internos controlados até espaços abertos com iluminação natural.

Dentre os trabalhos analisados, aquele que mais se assemelha ao presente estudo é o de Boom et al. (2017), que propõe um método para estimar a posição da fonte de luz utilizando sensores de profundidade, como o *Kinect*, combinados com imagens RGB. A abordagem deles permite estimar a posição 3D da luz e foi validada tanto em cenas sintéticas quanto em um conjunto de dados com medições reais, alcançando um erro médio de 20°. O método se mostrou adequado para renderização de objetos sintéticos com iluminação coerente, tornando-se uma solução relevante para aplicações de realidade aumentada.

Embora compartilhe o mesmo objetivo de estimar a iluminação para melhorar a integração entre objetos virtuais e o ambiente real, o método proposto neste trabalho apresenta diferenças fundamentais. A principal delas é que o sistema aqui desenvolvido não depende de sensores de profundidade, operando apenas com imagens RGB padrão, o que permite sua aplicação em dispositivos móveis comuns. Além disso, enquanto Boom et al. (2017) calculam a posição 3D da fonte de luz, o presente estudo foca na direção da iluminação, permitindo a projeção de sombras coerentes sem a necessidade de reconstrução geométrica detalhada. Essa abordagem reduz a complexidade computacional e viabiliza sua execução em tempo real, sem comprometer significativamente a precisão dos resultados.

## 4 Sistema Proposto

Neste capítulo é discutido o sistema desenvolvido para estimar o vetor de luz direcional em cenas de realidade aumentada, com foco em dispositivos móveis. O sistema opera inteiramente por meio de uma aplicação web, onde o usuário captura uma imagem da cena real. Essa imagem é então processada por um *pipeline* que combina técnicas de segmentação de imagem e geometria para calcular o vetor de luz da cena.

Para o funcionamento adequado do sistema, a imagem capturada deve conter alguns elementos essenciais. É necessário que haja um marcador fiducial posicionado em uma superfície plana, um ou mais objetos próximos ao marcador que projetem sombras bem definidas, e uma iluminação predominante, preferencialmente solar. Esses componentes permitem que os algoritmos de segmentação isolem os objetos e suas sombras, fundamentais para o cálculo do vetor de luz direcional.

Por outro lado, certas condições podem comprometer o desempenho do sistema. Iluminação difusa ou fraca dificulta a distinção entre objeto e sombra. Objetos translúcidos, com cores muito escuras ou texturas variadas podem ser erroneamente interpretados como sombras pelos algoritmos. Além disso, sombras que se sobrepõem aos próprios objetos ou a outros elementos do cenário também são um problema para a segmentação.

A partir dessa imagem, o *pipeline* de processamento realiza a segmentação, identifica os centros de massa do objeto e da sombra, e calcula a direção da luz utilizando técnicas geométricas e de renderização reversa. Os detalhes desse processo, bem como a arquitetura geral do sistema, são apresentados nas seções seguintes.

### 4.1 Arquitetura do Sistema

O sistema foi desenvolvido utilizando tecnologias acessíveis e amplamente utilizadas, com o objetivo de criar uma solução que pudesse ser executada diretamente em navegadores, sem a necessidade de servidores externos. Essa abordagem foi escolhida para garantir a acessibilidade do sistema, especialmente em dispositivos móveis.

Os algoritmos de segmentação utilizados não dependem de imagens de alta qualidade. Por isso, é possível a utilização do sistema em dispositivos com câmeras simples, como celulares e *webcams*. Isso se alinha com nosso objetivo inicial de acessibilidade.

Para o processamento de imagens, foi utilizada a biblioteca *OpenCV*, conhecida por sua ampla utilização na área de visão computacional. Essa ferramenta permitiu utilizar algoritmos como *GrabCut* e *K-means* para segmentar a imagem, isolando os objetos e suas sombras. Todo o processamento é realizado no navegador graças ao uso da biblioteca *Pyodide*, que possibilita a execução de código *Python* diretamente no ambiente *JavaScript*.

A interface web foi construída com *HTML*, *CSS* e *JavaScript*, enquanto as bibliotecas *Three.js* e *AR.js* foram utilizadas para renderizar objetos 3D e integrar a cena virtual com o ambiente real. A junção dessas ferramentas permitiu a criação das cenas em realidade aumentada diretamente no ambiente web.

O sistema foi hospedado no *GitHub Pages*, permitindo acesso direto à aplicação via navegadores. Essa solução simplificou o desenvolvimento e eliminou a necessidade de infraestrutura de servidores.

## 4.2 Funcionamento do Sistema

O sistema desenvolvido é estruturado no *pipeline* de processamento apresentado na Figura 4.1 e ilustrado na Figura 4.2. Cada etapa desempenha um papel fundamental no funcionamento geral do sistema. Por isso, cada uma dessas etapas será detalhada ao longo desta seção, destacando os métodos utilizados.

A entrada do sistema consiste em uma imagem contendo elementos essenciais, previamente descritos neste trabalho. Posteriormente, métodos geométricos são aplicados para associar as sombras aos objetos correspondentes. Por fim, utilizando técnicas de renderização inversa, o sistema calcula o vetor de luz direcional, permitindo a projeção de sombras virtuais alinhadas com as condições de iluminação da cena.

Cada uma dessas etapas será detalhada nas subseções a seguir, abrangendo desde a entrada de dados até o processamento geométrico e a renderização final.

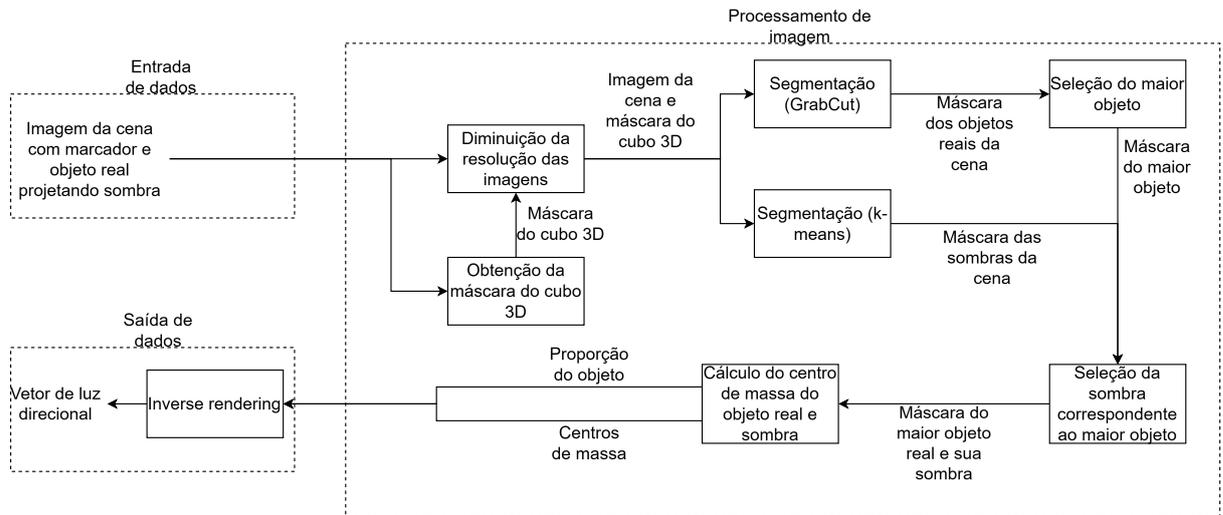
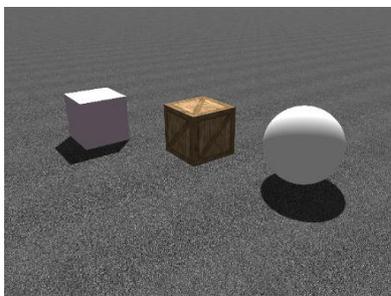
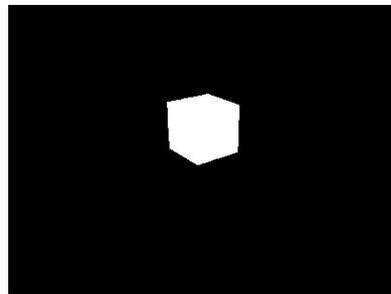


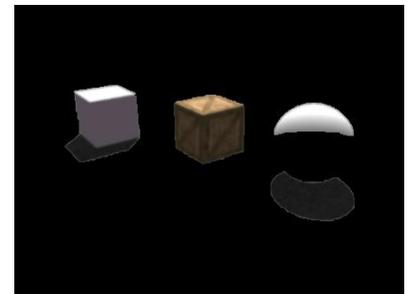
Figura 4.1: Diagrama do *pipeline* de processamento do sistema.



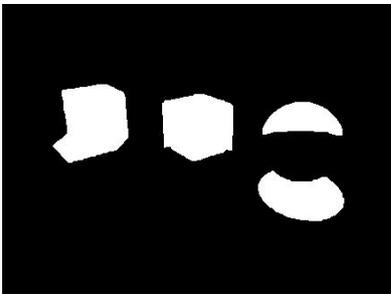
(a) Entrada - Imagem da cena



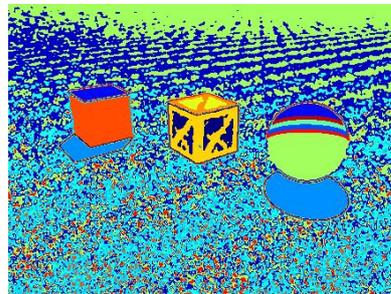
(b) Entrada - Máscara do cubo



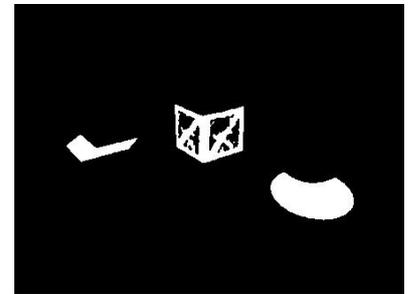
(c) Segmentação - GrabCut



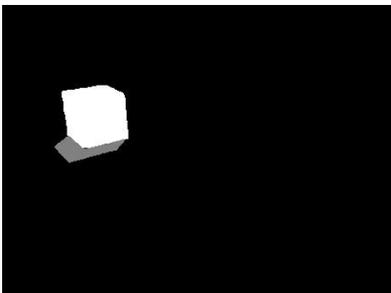
(d) Segmentação - Máscara do GrabCut



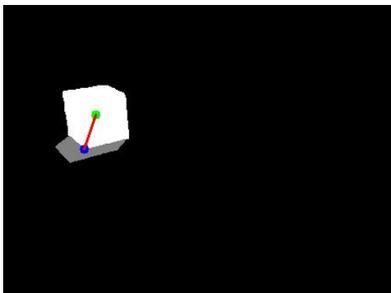
(e) Segmentação - Aplicação do K-means



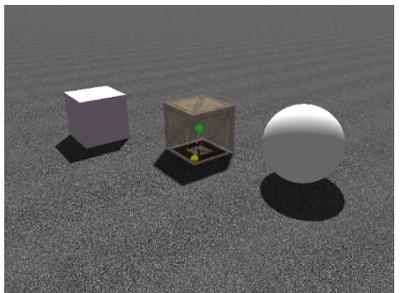
(f) Segmentação - Máscara K-means



(g) Maior objeto e sua respectiva sombra



(h) Centros de massa - Vetor 2D



(i) Inverse Rendering - Estimativa 3D

Figura 4.2: Visualização do *pipeline* do sistema.

### 4.2.1 Entrada de Dados

A entrada de dados do sistema é composta por duas imagens distintas que, em conjunto, fornecem as informações necessárias para o processamento. A primeira imagem é a captura da cena real, obtida por meio de uma câmera comum (Figura 4.2a). Essa imagem deve conter o marcador fiducial, os objetos reais próximos que projetam sombras e a fonte de luz predominante (preferencialmente solar). A segunda imagem é gerada a partir da renderização do objeto virtual via *AR.js*, que exibe o cubo com uma cor única, permitindo a extração precisa de sua posição e limites, resultando em uma máscara (Figura 4.2b).

A máscara do cubo é obtida a partir de uma renderização intermediária realizada em um *render target*, onde o cubo é destacado com um material de cor única. Essa máscara possibilita isolar o objeto virtual da cena.

Para otimizar o desempenho, a imagem capturada da cena é redimensionada para uma resolução de  $640 \times 480$  pixels, equilibrando a qualidade necessária para a análise com a redução do consumo de memória.

### 4.2.2 Segmentação de imagens

Foram utilizados dois algoritmos principais: *K-means*, empregado na detecção de sombras, e *GrabCut*, responsável pela segmentação dos objetos em primeiro plano.

O primeiro passo da segmentação consiste na detecção de sombras utilizando o *K-means*. O algoritmo agrupa os *pixels* da imagem em diferentes *clusters* com base em suas características de cor e luminosidade, conforme ilustrado na Figura 4.2e. No espaço de cor LAB, os *clusters* mais escuros são classificados como sombra. Como resultado, são geradas máscaras binárias onde os *pixels* das sombras são marcados em branco (255) e os demais em preto (0), como visto na Figura 4.2f. Para garantir que os resultados sejam replicáveis entre diferentes execuções, foi definida uma *seed* fixa, evitando variações indesejadas na segmentação.

A segmentação dos objetos em primeiro plano é realizada com o *GrabCut*. Inicialmente, foi definida uma região de interesse representada por um retângulo que abrange quase toda a imagem, estratégia que permite a automação do processo, dispensando a intervenção manual. A partir dessa região inicial, a segmentação é refinada iterativamente,

separando o primeiro plano do fundo, conforme mostrado na Figura 4.2c. O resultado final gerado pelo *GrabCut* é uma máscara binária onde os objetos segmentados aparecem isolados, como ilustrado na Figura 4.2d.

### 4.2.3 Associação entre Objeto e Sombra

Após a segmentação da imagem, o sistema gera dois conjuntos de máscaras binárias que representam objetos (Figura 4.2c) e sombras (Figura 4.2f). Para evitar que o objeto 3D virtual seja considerado na análise, sua máscara (Figura 4.2b) é aplicada para excluí-lo dos próximos passos do processamento. Dentre as máscaras de objetos, a de maior área é selecionada, pois o algoritmo *GrabCut* pode produzir pequenos artefatos indesejados. Esse critério assegura que a máscara selecionada corresponda a um objeto real, diminuindo a interferência de ruídos.

Após a seleção da maior região correspondente a um objeto, calcula-se seu centro de massa, que é definido como o ponto médio das posições de todos os *pixels* que compõem a região identificada. Sua determinação é realizada por meio das seguintes fórmulas:

$$cx = \frac{M_{10}}{M_{00}}, \quad cy = \frac{M_{01}}{M_{00}},$$

onde  $cx$  e  $cy$  correspondem às coordenadas horizontais e verticais do centro de massa,  $M_{00}$  é a área total da região, representada pela soma dos *pixels* que pertencem ao objeto, e  $M_{10}$  e  $M_{01}$  são os momentos espaciais relacionados à distribuição dos *pixels* em relação aos eixos horizontal e vertical.

Esse cálculo é aplicado tanto ao maior objeto quanto a cada sombra identificada. A partir disso, o sistema utiliza critérios baseados em distância e tamanho para determinar qual sombra pertence ao objeto principal. A distância euclidiana  $d$  é calculada entre o centro de massa do objeto e os centros de massa de cada sombra, utilizando a fórmula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2},$$

onde  $(x_1, y_1)$  são as coordenadas do centro de massa do objeto, e  $(x_2, y_2)$  as da sombra. Este critério assegura que sombras mais próximas do objeto sejam priorizadas.

Além disso, o sistema considera a área relativa das sombras, penalizando aquelas que são pequenas ou estão distantes do objeto. Essa penalização é implementada por meio de uma função exponencial, ajustando a área de acordo com a distância:

$$j = \frac{a}{\exp(d/c) + k},$$

onde  $j$  é a área ajustada,  $a$  é a área,  $d$  é a distância,  $c$  é uma constante de escala que controla o impacto da distância, e  $k$  evita divisões por zero. Com base na distância e na área ajustada, cada sombra recebe uma pontuação  $s$  calculada como:

$$s = \frac{d}{j + 1}$$

A sombra com a menor pontuação é selecionada como correspondente ao objeto principal. Esse método de associação é eficaz em cenários com sombras bem definidas e objetos claramente visíveis. No entanto, ele pode apresentar limitações em situações onde as sombras se sobrepõem ou são projetadas em superfícies irregulares. O resultado da associação do maior objeto com sua sombra é visível na Figura 4.2g do *pipeline*.

#### 4.2.4 Cálculo do Vetor de Luz Direcional

O cálculo do vetor de luz direcional é a etapa final do *pipeline* proposto, empregando técnicas de renderização inversa para determinar a direção da luz com base nos elementos extraídos da imagem bidimensional. Essa etapa combina informações geométricas da cena com ajustes específicos para correlacionar os dados processados da imagem 2D ao espaço tridimensional.

Para realizar esse cálculo, o sistema utiliza o centro de massa do maior objeto real para corresponder ao ponto central do modelo 3D gerado no marcador fiducial, transformando-o em referência no espaço tridimensional. Como o sistema não possui diretamente a sombra do modelo 3D, ele parte do princípio de que, se o objeto real tivesse a mesma proporção e tamanho do objeto virtual, suas sombras seriam equivalentes. Partindo dessa lógica, o sistema ajusta a escala do vetor entre o centro do objeto real e o centro de sua sombra (Figura 4.2h) de acordo com a proporção relativa entre o objeto

real e o virtual. Esse ajuste permite que a posição estimada da sombra do modelo virtual seja estimada, utilizando como referência as características do objeto real e sua sombra.

Após o ajuste de escala, o sistema realiza um *raycasting*, que consiste em lançar um feixe virtual a partir de um ponto de origem (como a posição da câmera) em direção a um alvo, como um ponto específico em um plano. No contexto do sistema proposto, essa técnica é usada para projetar o ponto 2D correspondente ao centro de massa da sombra do objeto real no plano tridimensional da cena virtual. O vetor calculado entre o centro do objeto real e o centro de sua sombra é escalado conforme o ajuste previamente calculado, garantindo que o ponto de interseção do *raycasting* forneça uma estimativa adequada da posição da sombra do modelo virtual em coordenadas 3D.

Com os pontos tridimensionais do centro geométrico do modelo virtual e da posição estimada da sombra projetada determinados, o vetor de luz direcional é calculado como a diferença vetorial entre essas duas coordenadas no espaço tridimensional. Esse vetor representa a direção da luz incidente na cena, que é então aplicada à iluminação da cena virtual, permitindo a geração de sombras coerentes com a iluminação do ambiente real. A Figura 4.2i é uma representação visual do vetor gerado (linha branca) entre o ponto central do objeto virtual (ponto verde) e o ponto de interseção do *raycasting* com o plano (ponto amarelo).

Além disso, o sistema oferece ao usuário a possibilidade de trocar os modelos virtuais associados ao marcador fiducial após o vetor de luz ser processado. Isso permite visualizar, em tempo real, como as sombras dos diferentes objetos virtuais se adaptam à luz calculada.

## 4.3 Resumo

Este capítulo apresentou o sistema desenvolvido para a estimativa do vetor de luz direcional em cenas de realidade aumentada, projetado para funcionar de forma acessível em dispositivos móveis por meio de uma aplicação web. Inicialmente, discutiu-se a entrada de dados, detalhando os elementos necessários para o funcionamento do sistema, como marcador fiducial, objetos com sombras bem definidas e iluminação predominante, além dos possíveis problemas na captura de imagens.

A seguir, foi descrita a arquitetura do sistema, destacando as tecnologias empregadas, como *OpenCV*, *Three.js* e *AR.js*, e o uso do *Pyodide* para integrar o processamento em *Python* diretamente no navegador. Essa abordagem permitiu criar uma solução independente da infraestrutura de servidores.

O *pipeline* de processamento foi explorado em detalhes, iniciando com a segmentação de imagem, onde algoritmos como *K-means* e *GrabCut* foram aplicados para identificar sombras e objetos. Posteriormente, abordou-se a associação entre objeto e sombra, empregando critérios geométricos baseados em distância e área ajustada para estabelecer as correspondências necessárias. Por fim, no cálculo do vetor de luz direcional, técnicas de renderização inversa e *raycasting* foram utilizadas para projetar elementos 2D no espaço tridimensional e determinar a direção da luz.

## 5 Resultados

Neste capítulo, os resultados obtidos pelo sistema são apresentados. Os testes foram realizados com imagens artificiais e reais. Porém, só foi possível obter a acurácia do algoritmo com as imagens artificiais, já que somente com elas é possível obter o ângulo de luz referencial da cena.

Os cenários reais foram obtidos por meio de fotografias capturadas com uma câmera de celular, utilizando uma impressão do marcador fiducial fixada ao solo e um objeto real posicionado próximo a ele. As imagens foram registradas ao ar livre, entre 10h e 15h, com incidência direta da luz solar para a formação de sombras bem definidas.

Já os cenários sintéticos foram gerados utilizando o *Three.js*, com cenas virtuais contendo uma imagem do marcador fiducial no centro e objetos geométricos simples ao redor. Essa abordagem permitiu a manipulação do vetor de luz direcional, possibilitando a criação de diferentes padrões de sombra na cena. Além disso, foram adicionadas texturas ao chão para simular irregularidades na superfície.

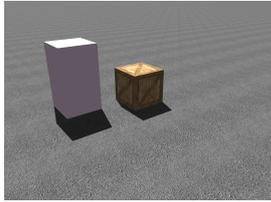
Os testes de desempenho foram realizados em dois dispositivos distintos. O primeiro é um dispositivo móvel, um iPhone 8 Plus com sistema operacional iOS 16, processador Apple A11 Bionic (6 núcleos,  $2 \times 2.39$  GHz Monsoon +  $4 \times 1.19$  GHz Mistral), GPU Apple de 3 núcleos, 3 GB de memória RAM. O segundo é um desktop com sistema operacional Windows 10, processador AMD Ryzen 7 5700X (8 núcleos, 16 threads, frequência base de 3.4 GHz), GPU AMD Radeon RX 6750XT e 32 GB de memória RAM.

### 5.1 Cenários Artificiais

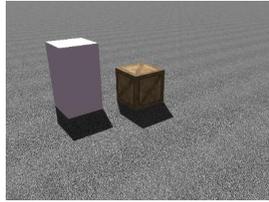
Os cenários artificiais permitiram a obtenção do vetor de luz direcional de cada cena, possibilitando a comparação direta entre os vetores calculados pelo sistema e os vetores reais utilizados para gerar a cena. Dessa forma, foi possível calcular o erro angular entre esses vetores e avaliar a acurácia da abordagem proposta.

Para a criação das cenas, buscou-se reproduzir condições ideais para o funciona-

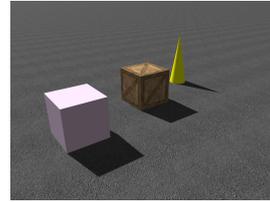
mento do sistema. Os objetos foram posicionados a uma distância adequada do marcador, evitando proximidade excessiva que pudesse comprometer a definição das sombras. Além disso, foram utilizados objetos com cores claras e fundos com padrões simples e pouco variados, reduzindo interferências nos algoritmos de segmentação.



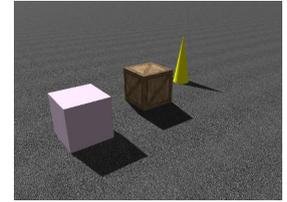
(a) Ground Truth - Cenário 1



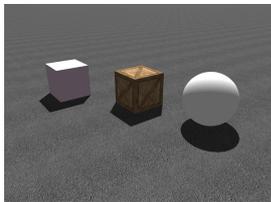
(b) Resultado Estimado - Cenário 1



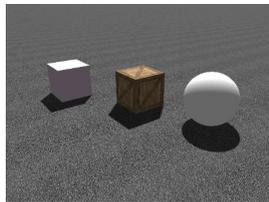
(c) Ground Truth - Cenário 2



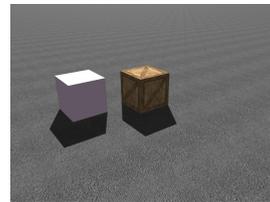
(d) Resultado Estimado - Cenário 2



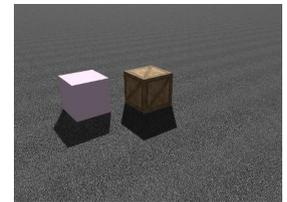
(e) Ground Truth - Cenário 3



(f) Resultado Estimado - Cenário 3



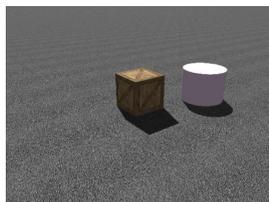
(g) Ground Truth - Cenário 4



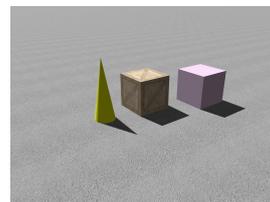
(h) Resultado Estimado - Cenário 4



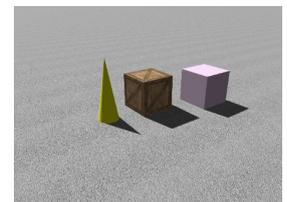
(i) Ground Truth - Cenário 5



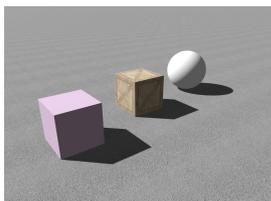
(j) Resultado Estimado - Cenário 5



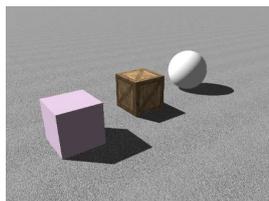
(k) Ground Truth - Cenário 6



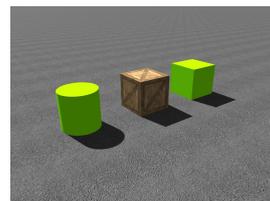
(l) Resultado Estimado - Cenário 6



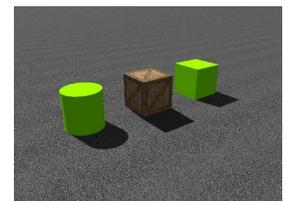
(m) Ground Truth - Cenário 7



(n) Resultado Estimado - Cenário 7



(o) Ground Truth - Cenário 8



(p) Resultado Estimado - Cenário 8

Figura 5.1: Comparação entre *Ground Truth* e Resultados Estimados em todos os cenários artificiais.

### 5.1.1 Resultados Quantitativos

Nesta seção, são apresentados os resultados quantitativos obtidos nos experimentos artificiais. O principal parâmetro avaliado foi o vetor de luz direcional estimado pelo sistema,

sendo sua precisão medida pelo erro angular em relação ao *ground truth*.

Na Figura 5.1, são apresentados os resultados dos testes realizados em cenários artificiais. As imagens estão organizadas em pares, com a primeira representando o *ground truth*, ou seja, o vetor de luz direcional gerado na cena virtual em *Three.js*, e a segunda ilustrando o vetor de luz estimado pelo sistema. Essa disposição permite a comparação visual entre os resultados.

Tabela 5.1: Erro angular entre os vetores reais e os vetores estimados.

Cenário	Vetor Real	Vetor Estimado	Erro Angular (°)
1	(1.15, 7, -4)	(-1.15, 1.42, -1.26)	16.12°
2	(-0.5, 3, -3)	(0.30, 1.35, -1.44)	15.47°
3	(4, 8, -4)	(0.95, 1.52, -0.88)	5.34°
4	(3, 5, -5)	(0.95, 1.19, -1.30)	5.80°
5	(3, 8, -5)	(0.45, 1.38, -1.37)	13.16°
6	(-1, 6, -5)	(-0.25, 1.40, -1.40)	5.15°
7	(-3, 4, -3)	(-0.46, 1.63, -1.06)	18.00°
8	(-1, 6, -5)	(0.14, 1.42, -1.41)	12.35°
<b>Média</b>	—		<b>11.42°</b>

Na Tabela 5.1 são apresentados os vetores reais e estimados para cada cenário, bem como os erros angulares associados. A média de 11.42° demonstra que o sistema apresenta uma precisão considerável, principalmente quando comparado ao pior caso possível, que seria um erro angular de 180°. O cálculo do erro angular segue a fórmula:

$$\theta = \cos^{-1} \left( \frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{|\mathbf{v}_1| |\mathbf{v}_2|} \right),$$

onde  $\mathbf{v}_1 \cdot \mathbf{v}_2$  é o produto escalar dos vetores e  $|\mathbf{v}_1|$  e  $|\mathbf{v}_2|$  são as magnitudes dos vetores. Assim, o erro angular representa a divergência direcional entre os vetores, expressa em graus, indicando o quão preciso é o algoritmo.

É possível observar a correlação entre os erros angulares presentes na Tabela 5.1, e os pares de imagens da Figura 5.1. Por exemplo, no cenário 7 (Figuras 5.1m e 5.1n), observa-se que a sombra do cubo estimada diverge significativamente da sombra real, refletindo o maior erro angular encontrado, 18.00°. Por outro lado, no cenário 6 (Figuras 5.1k e 5.1l), o erro angular foi o menor, 5.15°, com a sombra estimada praticamente alinhada à sombra real, tornando a diferença pouco perceptível.

Para o dispositivo móvel, a média de execução das imagens sintéticas foi de 6,73

segundos, enquanto para o desktop foi de 3,86 segundos, representando uma execução 42,65% mais rápida no desktop em comparação ao mobile. Para garantir uma avaliação mais precisa do desempenho, foram desconsideradas as primeiras execuções, uma vez que o carregamento inicial do *Pyodide* pode influenciar significativamente o tempo de processamento. Vale ressaltar que os tempos de execução são únicos, isto é, uma vez realizado o processamento, o vetor de iluminação da imagem permanece fixo, eliminando a necessidade de reprocessamentos futuros.

## 5.2 Cenários Reais

As imagens dos cenários reais foram capturadas com o objetivo de observar o funcionamento do sistema em condições não controladas. Diferentemente dos cenários artificiais, não há um vetor de iluminação de referência para comparação direta, impossibilitando a quantificação exata do erro angular.

No entanto, esses testes serviram para avaliar o funcionamento dos algoritmos de segmentação em ambientes complexos. As imagens permitiram verificar a capacidade do sistema de distinguir corretamente os objetos do fundo e identificar suas respectivas sombras, mesmo diante de variações na iluminação, formatos dos objetos, cores e texturas de fundo.



(a) Cenário real 1

(b) Cenário real 2

(c) Cenário real 3

Figura 5.2: Resultados em cenários reais com cubo texturizado virtual.

### 5.2.1 Resultados qualitativos

Nesta seção, são analisados qualitativamente os resultados obtidos nos cenários reais. Enquanto a seção anterior apresentou os valores numéricos dos erros angulares para os

cenários artificiais, aqui a avaliação se baseia na coerência visual das sombras projetadas, permitindo verificar o comportamento do sistema em condições não controladas.

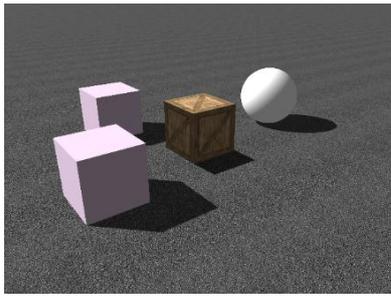
A Figura 5.2 apresenta os resultados para os cenários reais. Embora não seja possível calcular o erro angular devido à ausência de *ground truth*, a análise visual indica que as sombras projetadas mantêm uma boa correspondência com as sombras reais da cena. Por exemplo, na Figura 5.2a, mesmo com um *background* claro e de textura complexa, a sombra do cubo não destoa em relação à sombra do copo real. Da mesma forma, nas Figuras 5.2b e 5.2c, o *background* escuro não compromete a segmentação, permitindo que o sistema produza um resultado visualmente coerente.



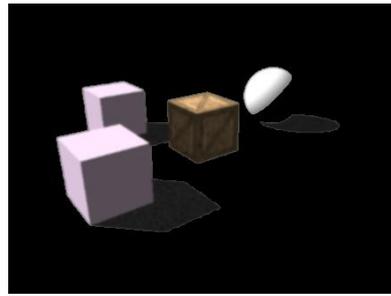
Figura 5.3: Visualização das cenas com diferentes objetos 3D.

Com o vetor de iluminação obtido, torna-se possível integrar diferentes objetos virtuais à cena, garantindo que suas sombras sejam projetadas corretamente, independentemente de suas formas. O sistema possibilita a substituição do cubo por outros modelos 3D, permitindo a visualização da integração de objetos mais complexos e realistas na cena, conforme ilustrado na Figura 5.3.

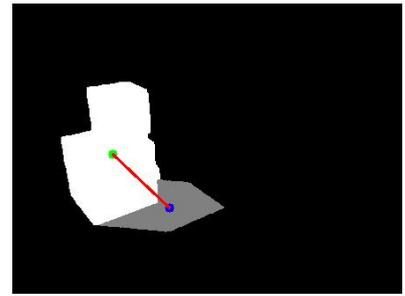
Além disso, observou-se que o sistema é capaz de lidar com objetos reais de diferentes formas e cores, bem como com variações na incidência da luz. Objetos com formas complexas, superfícies claras e sombras parcialmente projetadas sobre si não comprometeram significativamente a estimativa da iluminação. O método também demonstrou robustez em diferentes ângulos de captura, distâncias e horários do dia.



(a) Erro 1 - resultado da estimativa



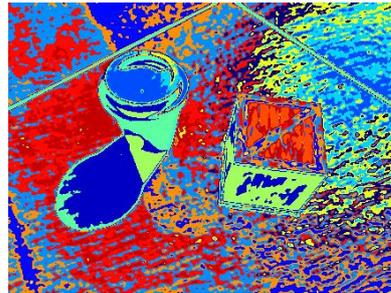
(b) Erro 1 - resultado segmentação grabcut



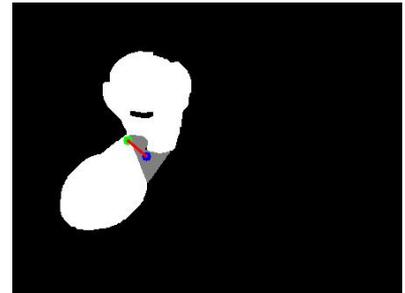
(c) Erro 1 - falha detecção de objeto



(d) Erro 2 - resultado da estimativa



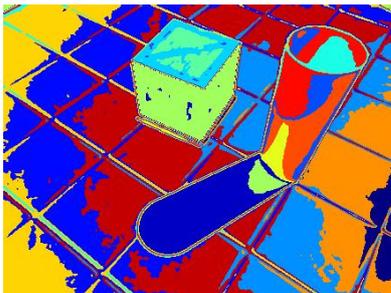
(e) Erro 2 - resultado segmentação K-means



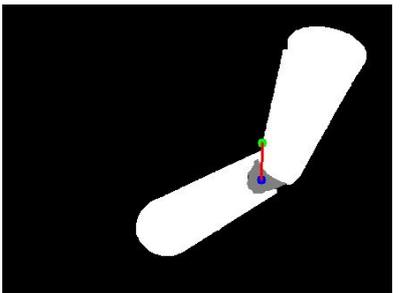
(f) Erro 2 - falha detecção de sombra



(g) Erro 3 - resultado da estimativa



(h) Erro 3 - resultado segmentação k-means



(i) Erro 3 - falha detecção de sombra

Figura 5.4: Erros do sistema.

## 5.3 Limitações do sistema

Apesar dos resultados apresentados nas seções anteriores, o sistema apresenta algumas limitações que podem impactar sua precisão e aplicabilidade em diferentes cenários. Durante a execução dos testes, foram identificadas limitações que afetam a estimativa do vetor de luz direcional, tornando necessário descartar algumas amostras que apresentaram falhas no processamento ou resultados muito fora do esperado.

Uma das principais limitações observadas foi a dificuldade do sistema em lidar com sombras e objetos sobrepostos ou parcialmente ocultos. O sistema assume que os objetos e suas sombras estão visíveis e isolados, mas, em casos onde múltiplos objetos e sombras se misturam ou são bloqueados por outros elementos da cena, a correspondência

entre o objeto e sua sombra se torna menos precisa. Este problema pode ser observado na Figura 5.4a, onde um cubo sobrepõe outro, dificultando o algoritmo de segmentação (Figura 5.4b). Este problema afeta diretamente a determinação do vetor de luz (Figura 5.4c), resultando em uma estimativa imprecisa da iluminação.

Outra limitação significativa é a sensibilidade do sistema à iluminação difusa. O método utilizado depende da presença de sombras bem definidas para estimar a direção da luz com precisão. No entanto, em ambientes com múltiplas fontes de iluminação ou luz difusa, como em dias nublados, as sombras podem perder contraste ou até se tornarem imperceptíveis, dificultando a segmentação correta. Esse problema é ilustrado nas Figuras 5.4d, 5.4e e 5.4f, onde a presença de duas fontes de luz resulta em sombras de intensidades diferentes. A superposição dessas sombras cria uma região de maior intensidade, que é erroneamente interpretada pelo algoritmo como sombra principal.

Além disso, o desempenho do sistema é afetado por superfícies com texturas muito variadas, tanto no plano de fundo quanto nos próprios objetos. Em cenários com padrões visuais complexos, superfícies reflexivas ou materiais translúcidos, os algoritmos de segmentação podem confundir partes do fundo com sombras ou até separar incorretamente regiões da sombra principal. Esse problema pode ser observado nas Figuras 5.4g, 5.4h e 5.4i, onde o objeto analisado é um copo levemente translúcido e reflexivo. Devido à sua translucidez, parte da luz atravessa o material, gerando uma sombra com intensidades variadas ao longo de sua projeção. Essas variações fazem com que a sombra projetada seja identificada de maneira inconsistente, resultando na segmentação da sombra em *clusters* distintos, o que compromete a identificação correta de sua forma e posição. Da mesma forma, padrões detalhados no plano de fundo podem ser erroneamente interpretados como objetos, interferindo no processo de segmentação e impactando a estimativa da iluminação.

Em termos de desempenho, o sistema foi projetado para rodar diretamente em navegadores nos dispositivos móveis. No entanto, a utilização de elementos de realidade aumentada pode ser computacionalmente intensiva, especialmente em dispositivos mais simples. Durante os testes, foi observado que, dependendo dos parâmetros utilizados nos algoritmos de segmentação, da complexidade dos modelos 3D ou da qualidade das

sombras, o tempo de processamento aumentava significativamente, podendo, em alguns casos, causar travamentos ou até mesmo a reinicialização da página.

Essas limitações não invalidam o funcionamento do sistema, mas demonstram que há espaço para melhorias e refinamentos. Algumas das restrições, como a dependência de sombras bem definidas, ocorrem pela abordagem deste trabalho, baseada em segmentação de imagem. Outras, como o desempenho, podem ser abordadas por meio da redução da qualidade da cena ou pela junção de mais técnicas de segmentação.

## 5.4 Resumo dos resultados

Os resultados apresentados no capítulo demonstraram que o sistema é eficaz na estimativa do vetor de luz direcional. Foram testados tanto cenários artificiais, nos quais foi possível calcular o erro angular, quanto cenários reais, nos quais a avaliação foi puramente qualitativa.

Nos testes com imagens sintéticas, o erro angular médio obtido foi de  $11.42^\circ$ , indicando uma precisão satisfatória na maioria dos casos. O menor erro registrado foi de  $5.15^\circ$ , enquanto o maior foi de  $18.00^\circ$ . Os melhores resultados foram obtidos em cenários mais simples, onde os objetos eram proporcionalmente próximos ao cubo virtual, enquanto ambientes com múltiplos objetos e ângulos complexos de sombra apresentaram maiores erros.

A análise qualitativa dos cenários reais evidenciou que o sistema é capaz de identificar corretamente os objetos e as sombras em cenários complexos, mantendo uma boa correspondência entre as sombras projetadas e as sombras reais da cena. A segmentação demonstrou eficiência mesmo em fundos variados, incluindo superfícies de texturas complexas e fundos escuros, sem comprometer significativamente o resultado final. Além disso, o sistema foi capaz de lidar com diferentes formas de objetos, cores e variações de iluminação, mantendo a consistência das sombras projetadas em diversos ângulos e horários do dia.

Durante os testes, algumas limitações foram identificadas, incluindo a sensibilidade à iluminação difusa, a dependência de cenários simples para segmentação eficiente e o impacto computacional da execução de RA em dispositivos móveis. Essas questões

---

indicam que há oportunidades para aprimoramentos, principalmente na robustez dos algoritmos de segmentação.

Os resultados obtidos reforçam a viabilidade da abordagem proposta e indicam que, mesmo com suas limitações, o sistema é capaz de realizar estimativas de luz direcional coerentes em cenários controlados, possibilitando a integração visual de objetos virtuais em cenas reais com maior realismo.

## 6 Conclusão

Neste trabalho, foi apresentado um sistema para estimar o vetor de luz direcional de uma cena 3D a partir de imagens retiradas de dispositivos simples, como celulares e *webcams*, utilizando técnicas geométricas e de segmentação. O objetivo principal foi permitir a integração mais realista de objetos virtuais em cenários reais, garantindo uma correspondência coerente entre a iluminação dos elementos sintéticos e do ambiente capturado.

O *pipeline* do sistema inicia com a captura de uma imagem contendo um marcador fiducial, sobre o qual um cubo virtual é posicionado. Em seguida, são aplicadas técnicas de segmentação para identificar objetos e sombras na cena. A partir dessas informações, o sistema estima o vetor de luz direcional utilizando a relação entre os centros de massa do objeto real e sua sombra. Posteriormente, são realizados ajustes de escala para correlacionar as proporções entre o objeto real e o virtual. Por fim, técnicas de renderização inversa são empregadas para transformar esse vetor obtido na imagem 2D em um vetor de luz tridimensional, utilizado na iluminação da cena.

Os experimentos realizados demonstraram que o sistema é capaz de estimar a direção da luz com um erro angular médio de  $11.42^\circ$ . Os testes em cenários artificiais confirmaram que a abordagem funciona bem em condições controladas. Já nos cenários reais, a avaliação qualitativa indicou que o sistema consegue gerar sombras coerentes com as sombras da cena, mesmo sob variações na posição da luz e na composição dos objetos. No entanto, a ausência de um *ground truth* para validação numérica impossibilitou uma análise quantitativa nesses casos.

Apesar dos resultados positivos, algumas limitações foram identificadas. A segmentação de imagem se mostrou o principal fator de erro do sistema, sendo sensível a fundos texturizados, sombras sobrepostas e variações na intensidade luminosa. Além disso, a presença de múltiplas fontes de luz ou sombras pouco definidas pode comprometer a precisão da estimativa. Em termos de desempenho computacional, o sistema apresentou uma execução eficiente na maioria dos dispositivos modernos, mas pode enfrentar dificuldades em hardware mais limitado, especialmente em dispositivos móveis mais antigos,

---

onde os cálculos de segmentação e renderização podem impactar a taxa de atualização da cena.

Ainda assim, o sistema demonstrou potencial para aplicações em realidade aumentada em cenários controlados. Os resultados obtidos indicam que abordagens geométricas podem ser viáveis para esse tipo de problema, sem a necessidade de técnicas mais complexas, como aprendizado de máquina. Como trabalhos futuros, melhorias na segmentação, estratégias para lidar melhor com cenários adversos e otimizações podem contribuir para aumentar a qualidade e aplicabilidade do sistema.

## Bibliografia

- AZINOVIC, D.; LI, T.-M.; KAPLANYAN, A.; NIESSNER, M. Inverse path tracing for joint material and lighting estimation. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2019. p. 2447–2456.
- AZUMA; RONALD, T. A survey of augmented reality. *Presence: Teleoperators & Virtual Environments*, MIT Press, v. 6, n. 4, p. 355–385, 1997.
- BOOM, B. J.; ORTS-ESCOLANO, S.; NING, X. X.; MCDONAGH, S.; SANDILANDS, P.; FISHER, R. B. Interactive light source position estimation for augmented reality with an rgb-d camera. *Computer Animation and Virtual Worlds*, Wiley Online Library, v. 28, n. 1, p. e1686, 2017.
- CAO, X.; FOROOSH, H. Camera calibration and light source orientation from solar shadows. *Computer Vision and Image Understanding*, Elsevier, v. 105, n. 1, p. 60–72, 2007.
- CHOTIKAKAMTHORN, N. Near point light source location estimation from shadow edge correspondence. In: IEEE. *2015 IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*. [S.l.], 2015. p. 30–35.
- GRUBER, L.; RICHTER-TRUMMER, T.; SCHMALSTIEG, D. Real-time photometric registration from arbitrary geometry. In: IEEE. *2012 IEEE international symposium on mixed and augmented reality (ISMAR)*. [S.l.], 2012. p. 119–128.
- KÁN, P.; KAFUMANN, H. Deeplight: light source estimation for augmented reality using deep learning. *The Visual Computer*, Springer, v. 35, n. 6, p. 873–883, 2019.
- KATO, H.; BEKER, D.; MORARIU, M.; ANDO, T.; MATSUOKA, T.; KEHL, W.; GAIDON, A. Differentiable rendering: A survey. *arXiv preprint arXiv:2006.12057*, 2020.
- KATO, H.; BILLINGHURST, M. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In: IEEE. *Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99)*. [S.l.], 1999. p. 85–94.
- KOC, E.; BALCISOY, S. Estimation of environmental lighting from known geometries for mobile augmented reality. In: IEEE. *2013 International Conference on Cyberworlds*. [S.l.], 2013. p. 132–139.
- LEGENDRE, C.; MA, W.-C.; FYFFE, G.; FLYNN, J.; CHARBONNEL, L.; BUSCH, J.; DEBEVEC, P. Deeplight: Learning illumination for unconstrained mobile mixed reality. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2019. p. 5918–5928.
- LIU, D. S.-M.; WU, S.-J. Light direction estimation and hand touchable interaction for augmented reality. *Virtual Reality*, Springer, v. 26, n. 3, p. 1155–1172, 2022.

- LOPEZ-MORENO, J.; GARCES, E.; HADAP, S.; REINHARD, E.; GUTIERREZ, D. Multiple light source estimation in a single image. In: WILEY ONLINE LIBRARY. *Computer Graphics Forum*. [S.l.], 2013. v. 32, n. 8, p. 170–182.
- MACQUEEN, J. Some methods for classification and analysis of multivariate observations. In: *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability/University of California Press*. [S.l.: s.n.], 1967.
- MARQUES, B. A. D.; CLUA, E. W. G.; MONTENEGRO, A. A.; VASCONCELOS, C. N. Spatially and color consistent environment lighting estimation using deep neural networks for mixed reality. *Computers & Graphics*, Elsevier, v. 102, p. 257–268, 2022.
- NGUYEN, R. M.; LE, M. N. Light source estimation from a single image. In: IEEE. *2012 12th International Conference on Control Automation Robotics & Vision (ICARCV)*. [S.l.], 2012. p. 1358–1363.
- OTSU, N. et al. A threshold selection method from gray-level histograms. *Automatica*, v. 11, n. 285-296, p. 23–27, 1975.
- ROTHER, C.; KOLMOGOROV, V.; BLAKE, A. "grabcut" interactive foreground extraction using iterated graph cuts. *ACM transactions on graphics (TOG)*, ACM New York, NY, USA, v. 23, n. 3, p. 309–314, 2004.
- SUTHERLAND, I. E. A head-mounted three dimensional display. In: *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*. [S.l.: s.n.], 1968. p. 757–764.
- WANG, Y.; SAMARAS, D. Estimation of multiple directional light sources for synthesis of augmented reality images. *Graphical Models*, Elsevier, v. 65, n. 4, p. 185–205, 2003.
- YU, Y.; SMITH, W. A. Inverserendernet: Learning single image inverse rendering. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2019. p. 3155–3164.