

**UNIVERSIDADE DO ESTADO DO AMAZONAS  
ESCOLA SUPERIOR DE TECNOLOGIA**

**JOEL DO CARMO NOGUEIRA**

**DESENVOLVIMENTO DE UM SISTEMA PARA DETECÇÃO DE  
OBJETOS DE UMA IMAGEM DE REFERÊNCIA ESTÁTICA  
UTILIZANDO MÉTODO ESTATÍSTICO PARAMÉTRICO PARA  
APLICAÇÕES DE CHROMA KEY**

MANAUS-AM  
2023

**JOEL DO CARMO NOGUEIRA**

**DESENVOLVIMENTO DE UM SISTEMA PARA DETECÇÃO DE  
OBJETOS DE UMA IMAGEM DE REFERÊNCIA ESTÁTICA  
UTILIZANDO MÉTODO ESTATÍSTICO PARAMÉTRICO PARA  
APLICAÇÕES DE CHROMA KEY**

Pesquisa desenvolvida durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia Elétrica da Escola Superior de Tecnologia da Universidade Estadual do Amazonas, como pré-requisito para a obtenção do título de Engenheira Eletricista.

Orientador: Jozias Parente de Oliveira.

Manaus  
2023

**Universidade do Estado do Amazonas – UEA**  
**Escola Superior de Tecnologia - EST**

*Reitor:*

**André Luiz Nunes Zogahib**

*Vice-Reitor:*

**Kátia do Nascimento Coureiro**

*Diretor da Escola Superior de Tecnologia:*

**Ingrid Sammyne Gadelha Figueiredo**

*Coordenador do Curso de Engenharia Elétrica:*

**Israel Gondres Torné**

*Banca Avaliadora composta por:*

**Prof. Dr. Jozias Parente de Oliveira (Orientador)**

**Prof. Dr. Fábio de Souza Cardoso**

**Prof. Dr. Israel Gondres Torne**

*Data da defesa: 28/08/2023.*

## **CIP – Catalogação na Publicação**

Nogueira, Joel do Carmo

Desenvolvimento de um sistema para detecção de objetos de uma imagem de referência estática utilizando método estatístico paramétrico para aplicações de Chroma Key/ Joel do Carmo Nogueira; orientado por Dr. Jozias Parente de Oliveira. – Manaus: 2023.  
65 p.: il.

Trabalho de Conclusão de Curso Graduação em Engenharia Elétrica.  
Universidade do Estado do Amazonas, 2023.

1. Processamento digital de imagens. 2. Método Estatístico Paramétrico. 3. Chroma Key.

I. Oliveira, Jozias Parente de.

JOEL DO CARMO NOGUEIRA

DESENVOLVIMENTO DE UM SISTEMA PARA DETECÇÃO DE OBJETOS DE UMA  
IMAGEM DE REFERÊNCIA ESTÁTICA UTILIZANDO MÉTODO ESTATÍSTICO  
PARAMÉTRICO PARA APLICAÇÕES DE CHROMA KEY

Pesquisa desenvolvida durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia Elétrica da Escola Superior de Tecnologia da Universidade Estadual do Amazonas, como pré-requisito para a obtenção do título de Engenheira Eletricista.

Nota obtida: 9,8 (nove vírgula oito)

Aprovado em 28 / 08 / 23.

Área de concentração: Processamento de Digital de Imagens

BANCA EXAMINADORA

Jozias P. de Oliveira  
Orientador: Jozias Parente de Oliveira, Dr.

Fábio de Souza Cardoso  
Avaliador: Fábio de Souza Cardoso, Dr.

Israel Gondres Torne  
Avaliador: Israel Gondres Torne, Dr.

## RESUMO

O tradicional método do Chroma Key utiliza uma técnica denominada *Key Mean Clustering* que exige uma considerável quantidade de armazenamento por meio dos *buffers*, bem como velocidade e complexidade do sistema para implementação em hardware. O presente projeto de pesquisa apresenta a validação em MATLAB do desenvolvimento de um sistema de baixa complexidade de implementação para as aplicações em Chroma Key. Este sistema realiza a extração de objetos de uma imagem de referência estática com fundo verde utilizando uma segmentação de imagem baseada em um modelo estatístico paramétrico que utiliza uma imagem média das componentes cores como referência para realizar a classificação de cada elemento de imagem. Para validação final foram utilizadas 120 imagens para testes, sendo 37 fotos do fundo verde e 83 fotos com a presença do objeto a ser detectado. Além disso, a avaliação de desempenho foi realizada utilizando-se as seguintes métricas: precisão, revocação, medida F e fator de confiança. Utilizando-se o plano de cores RGB, obteve-se uma acurácia máxima de 100% e mínima igual a 99,4%. No caso do plano de cores YCbCr a acurácia máxima obtida foi igual a 100% e a mínima igual a 99,9%. Realizando-se o teste de inferência estatística, conclui-se que não há diferenças estatisticamente significantes entre os dois planos de cores.

**Palavras-chave:** Processamento Digital de Imagens. Método Estatístico Paramétrico. Chroma Key.

## ABSTRACT

The traditional Chroma Key method employs a technique called Key Mean Clustering, which demands a substantial amount of storage through buffers, as well as system speed and complexity for hardware implementation. This research project presents the MATLAB validation of the development of a low-complexity implementation system for Chroma Key applications. This system performs the extraction of objects from a static reference image with a green background using image segmentation based on a parametric statistical model. This model utilizes an average image of color components as a reference to classify each image element. For final validation, 120 images were used for testing, consisting of 37 photos with a green background and 83 photos with the object to be detected present. Additionally, performance evaluation was conducted using the following metrics: precision, recall, F-measure, and confidence factor. Utilizing the RGB color space, a maximum accuracy of 100% and a minimum of 99.4% were achieved. In the case of the YCbCr color space, the maximum accuracy attained was 100%, and the minimum was 99.9%. Through statistical inference testing, it can be concluded that there are no statistically significant differences between the two color spaces.

**Keywords:** Digital Image Processing; Parametric Statistical Method; Chroma Key.

## LISTA DE ILUSTRAÇÕES

Figura 1: Técnicas de processamento de Imagens aplicadas à imagem .....	13
Figura 2: Representação de uma Imagem Digital Bidimensional. ....	14
Figura 3: Processo de discretização de uma imagem digital. ....	15
Figura 4: Comparação do Efeito da Variação da Quantização em um Imagem.....	15
Figura 5: Representação do Espectro de Luz Visível.....	16
Figura 6: Representação do Sistema RGB no Espaço Tridimensional. ....	17
Figura 7: Representação da Conversão do Sistema de Cor RGB para YCbCr. ....	18
Figura 8: Fluxograma do Processamento e Análise de Imagem Digital. ....	19
Figura 9: Limiarização de uma Imagem: (a) Histograma Original e (b) Histograma da Imagem Binarizada.....	20
Figura 10: Representação do Modelo Estatístico Paramétrico. ....	22
Figura 11: Aplicação de Chroma Key nas previsões do tempo nos Telejornais. ....	24
Figura 12: Descrição do processo da técnica de <i>Chroma Key</i> . ....	24
Figura 13: <i>Chroma Key</i> na indústria do Cinema. ....	25
Figura 14: Etapas da Metodologia para o Desenvolvimento da Pesquisa. ....	27
Figura 15: Exemplo de sequências de imagens para avaliação do método de segmentação proposto – (a) imagem de teste; (b) imagem de teste de referência (ground truth).....	29
Figura 16: Leitura das Imagens de Fundo de Referência e Cálculo da Média de Cada Elemento de Imagem. ....	33
Figura 17: Quadros de Imagens – Fundo Verde.....	33
Figura 18: Trecho Algoritmo - Leitura das Imagens de Testes.....	34
Figura 19: Imagens de Testes com Objetos com Fundo Verde (Testes Iniciais). ....	34
Figura 20: Definição dos Valores de Limiares e Separação das Componentes de Cores .....	35
Figura 21: Detecção do Objeto e Segmentação (Trechos do Algoritmo -APÊNDICE A).....	35
Figura 22: Resultado Segmentação RGB Limiar (R=50; G=50; B=50) .....	36
Figura 23: Comparação de Seleção de Diferentes Limiares RGB. ....	36
Figura 24: Criação Cenário de Chroma Key .....	38
Figura 25: Ajuste de Foco e Luminosidade Automática. ....	38
Figura 26: Resultado Segmentação RGB Imagens Estáticas Reais com Diferentes Limiares. ....	39
Figura 27: Resultados da Segmentação em RGB (Sem ajuste da Câmera de Vídeo). ....	40
Figura 28: Parâmetros de Configurações da Câmera. ....	41

Figura 29: Trecho Algoritmo - Ajustes e Configurações da Câmera .....	41
Figura 30: Resultados da Segmentação em RGB (Com ajuste da Câmera de Vídeo). .....	42
Figura 31: Imagens de Testes com suas Respectivas Referências. ....	43
Figura 32: Cálculo das métricas e do Fator de Confiança. ....	43
Figura 33: Comparação dos Resultados da Segmentação para RGB e YCbCr.....	44
Figura 34: Gráfico de Desempenho dos Algoritmos nos Sistemas de Cores RGB e YCbCr – Métrica Precisão. ....	44
Figura 35: Gráfico de Desempenho dos Algoritmos nos Sistemas de Cores RGB e YCbCr – Métrica Revocação. ....	45
Figura 36: Gráfico de Desempenho dos Algoritmos nos Sistemas de Cores RGB e YCbCr – Métrica Medida F. ....	45
Figura 37: Preenchimento de Hipótese Inicial e Dados .....	46
Figura 38: Gráfico Distribuição t Student. ....	46
Figura 39: Resultado Final da Segmentação com aplicação Chroma Key .....	47



## SUMÁRIO

<b>INTRODUÇÃO</b> .....	<b>11</b>
<b>1 REFERENCIAL TEÓRICO</b> .....	<b>13</b>
1.1 PROCESSAMENTO DIGITAL DE IMAGENS .....	13
<b>1.1.1 Fundamentos de uma Imagem Digital</b> .....	<b>14</b>
<b>1.1.2 Amostragem e Quantização</b> .....	<b>15</b>
<b>1.1.3 Sistemas de Cores</b> .....	<b>16</b>
<b>1.1.4 Sistema RGB</b> .....	<b>16</b>
<b>1.1.5 Sistema YCbCr</b> .....	<b>17</b>
1.2 SEGMENTAÇÃO .....	18
<b>1.2.1 Limiarização (Thresholding)</b> .....	<b>20</b>
<b>1.2.2 Modelo Paramétrico e não-Paramétrico</b> .....	<b>21</b>
<b>1.2.3 Modelo Estatístico Paramétrico (Gaussiano)</b> .....	<b>21</b>
<i>1.2.3.1 Valor Médio</i> .....	<i>22</i>
<i>1.2.3.2 Desvio Padrão</i> .....	<i>23</i>
<i>1.2.3.3 Classificação dos pixels</i> .....	<i>23</i>
1.3 APLICAÇÃO CHROMA KEY .....	23
<b>2 METODOLOGIA</b> .....	<b>26</b>
2.1 AMBIENTE DE DESENVOLVIMENTO.....	26
2.2 MATERIAIS .....	26
2.3 MÉTODOS.....	27
<b>2.3.1 Revisão Bibliográfica</b> .....	<b>27</b>
<b>2.3.2 Definição do método de Segmentação e Criação de Imagens Digitais (Testes Iniciais)</b> .....	<b>27</b>
<b>2.3.3 Desenvolvimento em MATLAB (Validação Inicial)</b> .....	<b>28</b>
<b>2.3.4 Métricas de Avaliação</b> .....	<b>28</b>
<b>2.3.5 Criação do Cenário de Testes e Ajustes da Câmera</b> .....	<b>30</b>
<b>2.3.6 Testes e Avaliação de Desempenho</b> .....	<b>31</b>
<b>3 IMPLEMENTAÇÃO E ANÁLISE DOS RESULTADOS</b> .....	<b>32</b>

3.1 DESENVOLVIMENTO DO ALGORITMO EM MATLAB E VALIDAÇÃO INICIAL...	32
3.2 CRIAÇÃO DO CENÁRIO CHROMA KEY E TESTES COM IMAGENS ESTÁTICAS REAIS.....	37
3.3 SEGMENTAÇÃO EM TEMPO REAL COM E SEM AJUSTES DA CÂMERA DE VÍDEO.....	39
3.4 MÉTRICAS DE AVALIAÇÃO DE DESEMPENHO E INFERÊNCIAS ESTATÍSTICAS .	42
CONCLUSÃO .....	48
REFERÊNCIAS BIBLIOGRÁFICAS .....	50
APÊNDICE A – ALGORITMO PARA VALIDAÇÃO INICIAL DO MÉTODO DE SEGMENTAÇÃO USANDO A MÉDIA .....	53
APÊNDICE B – ALGORITMO DE SEGMENTAÇÃO (MÉDIA) USANDO UMA CÂMERA DE VÍDEO .....	56
APÊNDICE C –SEGMENTAÇÃO DE IMAGEM NO SISTEMA DE COR YCBCR COM MÉTRICAS DE AVALIAÇÃO .....	59
APÊNDICE D – TABELAS DAS INFERÊNCIAS ESTATÍSTICAS (DISTRIBUIÇÃO T STUDENT) .....	64

## INTRODUÇÃO

O processamento digital de imagens (PDI) tem evoluído constantemente à medida que a tecnologia avança, contribuindo para uma melhoria na extração de informações presentes em imagens. Muitas atividades do cotidiano que requerem a análise ou extração de objetos de uma imagem podem ser desenvolvidas com o auxílio de técnicas PDI, como exemplo, áreas médicas – diagnósticos de doenças pelas análises de imagens de raios-X baseadas em diferenças de texturas (SILVA, 2014).

A segmentação de imagens é a etapa crucial na análise de imagens e aplicações em visão computacional. Segundo (SCHADE, 2019), define a segmentação de imagens como o particionamento de uma imagem em diferentes aglomerados ou seções de acordo com a intensidade do pixel. A segmentação tem sido uma etapa inevitável na maioria das técnicas para a detecção e classificação de objetos. Com o crescimento exponencial da tecnologia da informação aliada com o auxílio de processadores computacionais tem tornado a segmentação popular em muitos campos, como por exemplo, medicina, segurança, multimídia e etc. A maioria desses sistemas usam a segmentação como um estágio de pré-processamento para localizar as regiões relevantes da imagem bruta (SCHADE, 2019).

O pré-processamento tem grande relevância para aplicações no campo da multimídia. Uma técnica muito comum que faz uso desta segmentação é a *chroma keying*, que é uma prática específica usada na edição de imagens e vídeos, como por exemplo, produção de filmes (LESTARI, 2018).

*Chroma Key* é uma importante e robusta técnica para processamento de imagem ou vídeo em que é amplamente aplicada nas indústrias de cinemas, vídeo game, bem como, nas clássicas previsões do tempo dos telejornais. Esta técnica realiza a extração de objetos do quadro de imagem e combina-os com um fundo qualquer a fim de compor novas cenas e criar efeitos especiais. No quadro em que os objetos são segmentados é composto de duas partes que consistem em objetos de primeiro e segundo plano. Sendo os objetos de segundo plano uma cor sólida, normalmente verde ou azul. E os objetos de primeiro plano são os objetos colocados em frente desta tela verde ou azul (TIWARI, BASKER, KALAIARASI, 2014).

O tradicional método do *Chroma Key* utiliza uma técnica denominada *Key Mean Clustering* que exige uma considerável quantidade de armazenamento por meio dos buffers que exigem maior velocidade e complexidade do sistema para implementação em hardware (TAI, 2014).

Desta maneira, surgiu-se a hipótese se seria possível o desenvolvimento de um sistema para a detecção de objetos para aplicação em *chroma key* com baixa complexidade na implementação.

Dessa forma, o presente projeto de pesquisa apresenta um sistema de baixa complexidade de implementação para aplicações em *chroma key*. Este sistema realiza a extração de objetos de uma imagem de referência estática (Fundo azul ou verde) utilizando uma segmentação baseada em um modelo estatístico paramétrico, para serem utilizados nessas.

Além disso, a pesquisa visa gerar benefícios na indústria do entretenimento e no ramo da multimídia, com objetivo de agregar ou aprimorar trabalhos e serviços que necessitem da presente tecnologia em estudo.

Dos objetivos e benefícios supracitados, o desenvolvimento deste projeto no âmbito acadêmico proporcionou as seguintes contribuições, quais sejam: revisão do estado da arte de algoritmos para segmentação de objetos e a ampliação e complementação dos conhecimentos teóricos adquiridos em sala de aula, nas seguintes disciplinas: Linguagem de programação I e II; Processamento Digital de Sinal e Imagem e rotinas e programação de algoritmos em MATLAB.

Para ordenação dos assuntos abordados, este trabalho está dividido em quatro capítulos, que são:

Capítulo 1 – Referencial Teórico: Neste capítulo são apresentados os conceitos fundamentais de Processamento Digital de Imagens e *Chroma Key*.

Capítulo 2 – Método Proposto: Neste capítulo é definido o método científico que foi utilizado para a condução da pesquisa.

Capítulo 3 – Implementação: Descrição, em detalhes, da implementação e validação em MATLAB.

Capítulo 4 – Testes e Resultados Obtidos: Descrição dos testes realizados e os resultados obtidos decorrentes das atividades realizadas em software.

Ao final, são apresentadas as conclusões finais do trabalho, com os principais fatos, aprendizados e dificuldades encontradas durante a realização do projeto.

## 1 REFERENCIAL TEÓRICO

Para este projeto de pesquisa, o método de segmentação utilizado foi uma técnica de segmentação baseada em um modelo estatístico paramétrico. Com isso, neste capítulo será feita uma revisão da literatura sobre os princípios básicos da área de processamento digital de imagem, bem como um estudo sobre a técnica de aplicação de Chroma Key e métricas de avaliação do resultado obtido. Tais tópicos serão abordados a seguir.

### 1.1 PROCESSAMENTO DIGITAL DE IMAGENS

O processamento digital de imagens possui grande importância para muitas áreas de pesquisas que requerem melhorias no visual de imagens ou que necessitem extrair informações ou características importantes para análises ou processamento de dados da imagem. O processamento de imagens engloba diversas abordagens, como por exemplo, aperfeiçoamento de imagens, usando técnicas de restauração, suavização, realce e segmentação de imagens (SILVA, 2014). Na **Figura 1**, tem-se um exemplo de aplicação do processamento digital de imagem, usado como melhoria de imagem:

**Figura 1:** Técnicas de processamento de Imagens aplicadas à imagem



**Fonte:** Silva, 2014.

### 1.1.1 Fundamentos de uma Imagem Digital

Define-se imagem monocromática como uma função bidimensional contínua  $f(x,y)$ , sendo  $x$  e  $y$  coordenadas espaciais, ou seja, largura e altura da imagem respectivamente. O valor de  $f$  para qualquer ponto  $(x,y)$  é proporcional à intensidade luminosa de brilho ou nível de cinza naquele ponto (QUEIROZ, 2001).

Pode-se representar a intensidade de luz como:

$$f(x, y) = i(x, y) * r(x, y) \quad (1)$$

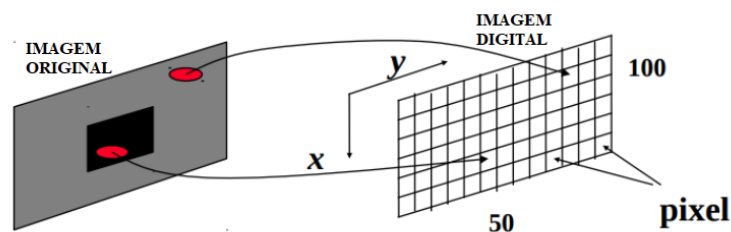
Sendo:

$i$ : iluminação do ambiente:  $0 < i(x, y) < \infty$

$r$ : reflectância dos ambientes:  $0 \leq r(x, y) \leq 1$

Dessa maneira, uma imagem digital pode ser representada em forma de matrizes bidimensionais conforme mostra **Figura 2**. Cada elemento dessa matriz representa um pixel da imagem. O pixel é associado a um valor numérico que representa sua intensidade luminosa. Para imagens em tons de cinza (escala de cinza), a intensidade geralmente é representada por um valor único em uma escala de 0 a 255, onde 0 representa preto e 255 representa branco. Valores intermediários representam tons de cinza entre o preto e o branco.

**Figura 2:** Representação de uma Imagem Digital Bidimensional.



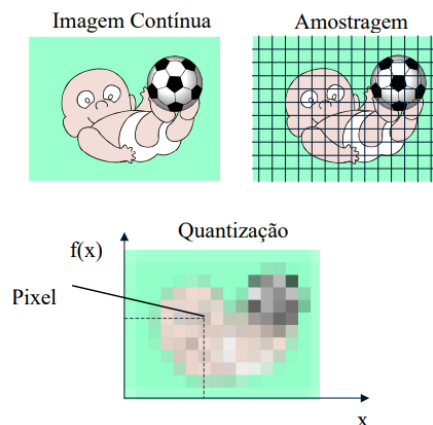
Fonte: Adaptado de ARAKI [s.d.].

### 1.1.2 Amostragem e Quantização

Para o armazenamento ou o processamento de uma imagem em um dispositivo eletrônico, é necessário a discretização tanto em nível de coordenadas espaciais quanto de valores de brilho (QUEIROZ, 2001).

Pode-se definir a amostragem como processo de discretização das coordenadas espaciais da imagem, ou seja, refere-se ao número de pontos amostrados de uma imagem digitalizada (resolução). A quantização é o processo de discretização dos valores de brilho da imagem, ou seja, quantidade de níveis de tons que pode ser atribuído a cada ponto digitalizado. Pode-se observar na **Figura 3**, o processo de discretização de uma imagem digital.

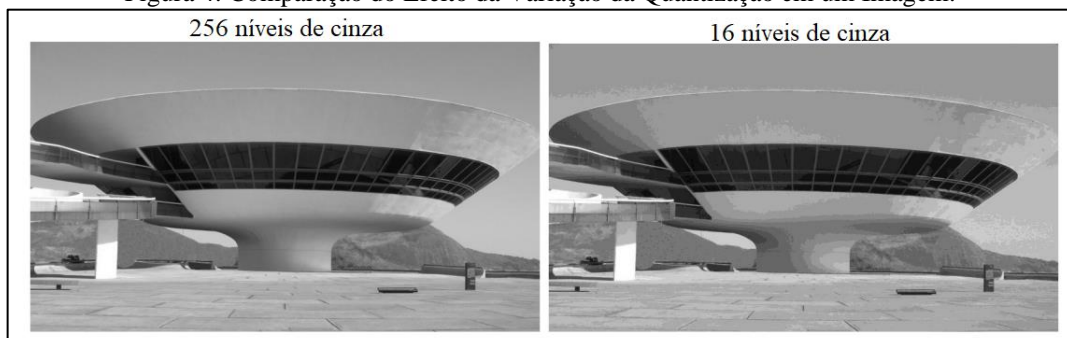
**Figura 3:** Processo de discretização de uma imagem digital.



Fonte: SCURI, 2002.

A quantização é realizada ao atribuir a cada pixel um valor inteiro dentro de uma escala pré-definida (por exemplo, de 0 a 255 para uma imagem em 8 bits). Isso reduz o número de níveis de intensidade e discretiza a imagem, tornando-a adequada para armazenamento digital. Na **Figura 04**, tem-se uma representação do efeito da variação da quantização na qualidade da imagem.

**Figura 4:** Comparação do Efeito da Variação da Quantização em um Imagem.



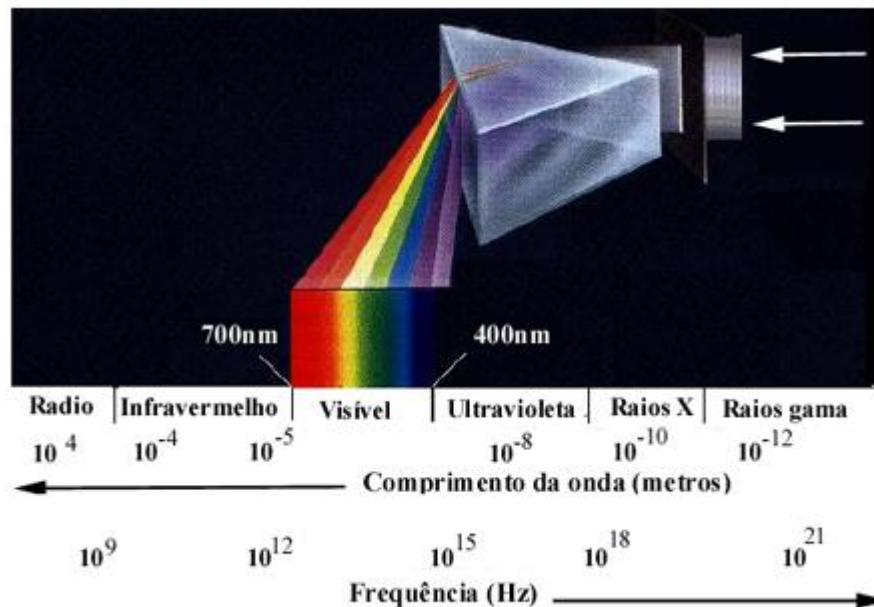
Fonte: Adaptado de ARAKI [s.d.].

### 1.1.3 Sistemas de Cores

A luz tem um comportamento ondulatório que é representado por uma frequência ( $f$ ) e um comprimento de onda ( $\lambda$ ). A luz visível encontra-se na faixa do espectro eletromagnético de 400 a 770 nm aproximadamente, logo qualquer valor fora desta faixa não é percebida pelo olho humano (QUEIROZ, 2001). Na **Figura 5**, tem-se uma representação do espectro de luz visível.

Em 1666, Isaac Newton descobriu que a luz branca ou a luz natural do sol pode ser decomposta em várias outras cores ao incidir sobre um prisma, podendo as faixas de cores variarem do violeta ao vermelho (MARTINS, 2015).

**Figura 5: Representação do Espectro de Luz Visível.**



Fonte: NISHIDA, [s.d.].

O sistema ou modelo de cores pode ser considerado como uma abstração de um modelo matemático para representar as cores presentes em um espectro de luz. Normalmente, esse modelo é tridimensional, em que cada dimensão representa uma componente, e a cor final é resultante da combinação de cada componente (SILVA, 2014).

### 1.1.4 Sistema RGB

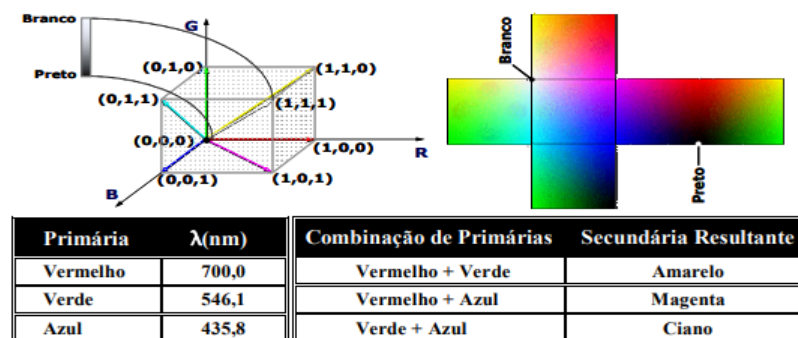
O modelo cromático RGB é obtido a partir do processo de geração, denominado aditivo. O processo aditivo consiste basicamente na combinação variável em proporção de componentes monocromáticas nas faixas espectrais associadas às sensações de cor verde, vermelho e azul,



sendo estas responsáveis pela criação das demais sensações de cores. Ou seja, as cores verde, vermelha e azul são chamadas de cores primárias e a partir da combinação entre as cores primárias obtêm-se as cores secundárias (QUEIROZ, 2001).

Na **Figura 6**, tem-se uma representação do sistema RGB no espaço de coordenadas cartesianas tridimensional, onde há o destaque das cores primárias e secundárias deste sistema.

**Figura 6:** Representação do Sistema RGB no Espaço Tridimensional.



Fonte: QUEIROZ, 2001.

Silva (2014) apresenta uma representação do sistema de cores RGB. Neste esquema mostra a cor preta sendo representada pela origem e a cor branca representada pelo ponto mais distante do preto.

Em um espaço de cores RGB, cada canto é associado a um valor numérico que representa a intensidade de vermelho, verde e azul. A origem é representada pela cor preta, enquanto o ponto mais distante do preto, ou seja, da origem, corresponde à cor branca. Entre esses extremos, há uma progressão de tons de cinza ao longo de uma linha que os conecta. As cores, por sua vez, são representadas por pontos localizados no interior de um cubo e são definidas por vetores que partem da origem (SILVA, 2014).

### 1.1.5 Sistema YCbCr

Esse modelo de cor é comumente utilizado em vídeo digital. A informação de luminância é representada pela componente Y, enquanto que a informação de cor é armazenada nas componentes Cb e Cr. A componente Cb representa a diferença entre a componente azul e um valor de referência e a Cr é a diferença entre a componente vermelha e um valor de referência (SANTOS, 2012). Na **Figura 7**, tem-se uma representação da conversação do sistema de cor RGB para YCbCr.

**Figura 7:** Representação da Conversão do Sistema de Cor RGB para YCbCr.

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.5 \\ 0.5 & -0.419 & -0.081 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Fonte: Santos, 2012.

## 1.2 SEGMENTAÇÃO

O processamento de imagens (PI) é uma das áreas da visão computacional que possui grande potencial de aplicabilidade, compreendendo um conjunto de técnicas úteis que melhoram a extração de informação a partir de determinadas imagens. O PI visa melhorar e realçar a informação presente nas imagens originais. Normalmente, a análise de imagem é representada pela segmentação, que corresponde na divisão da imagem da mesma em sub-regiões que a constituem (SILVA, 2014).

A segmentação consiste em dividir uma imagem em suas unidades significativas a fim de obter os objetos de interesse que compõem uma imagem (FILHO, 1999).

Segundo Morgan (2008), define segmentação de imagem como:

Um processo fundamental para extração e identificação de objetos ou áreas de interesse da imagem, bem como permite reduzir a informação da mesma a fim de manipular objetos simples de uma cena que geralmente correspondem a linhas ou regiões (grupos de pontos conectados) de interesse.

O processo de segmentação de imagens é considerado uma das etapas necessárias para a obtenção de informações qualitativas, pois garante a identificação do objeto de interesse. E além disso, o processo gera informações quantitativas, como exemplo, área, perímetro e volume. (CHAGAS, 2009).

Segundo Augusto (2012), define o processamento de imagem como se fossem a composição de duas técnicas:

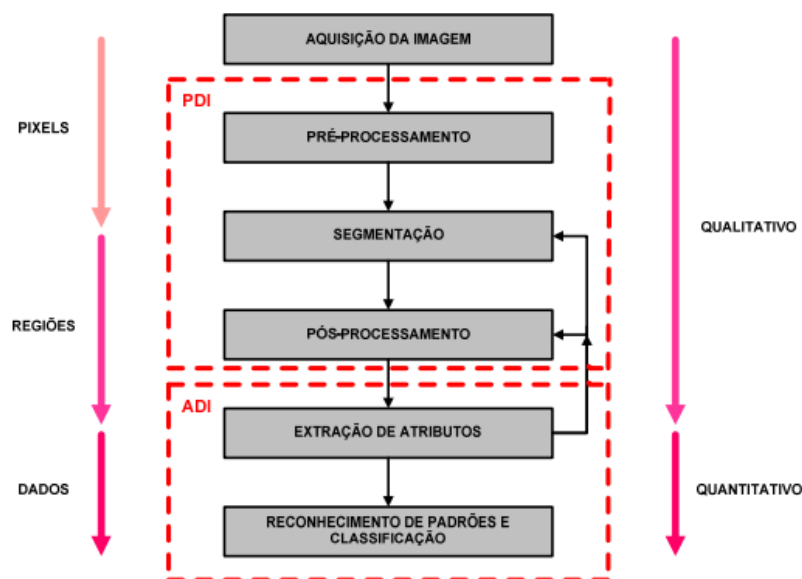
- a) Processamento Digital de Imagens (PDI): preparação inicial da imagem para análises futuras, como por exemplo, operações matemáticas que alteram os valores dos pixels.
- b) Análise Digital de Imagens (ADI): consiste na análise quantitativa do processo em que regiões, partículas e objetos identificados na imagem são medidos.

Na **Figura 8**, tem-se uma representação do fluxograma do processamento e análise digital de imagens. Pode-se observar que o estágio inicial do processo é a aquisição da imagem, com a formação e a digitalização da imagem. Com a obtenção da imagem digital, a próxima

etapa consiste no pré-processamento da imagem digitalizada, nesta etapa tem o objetivo de realizar uma melhoria da imagem para que próximas sejam realizadas com sucesso. A segmentação se iniciará após a finalização do pré-processamento, neste momento ocorrerá a separação da imagem em partes ou objetos de interesse. Augusto (2012), ainda ressalta que o resultado da segmentação pode não ser adequado como esperado, e com isso, a etapa de pós-processamento é executada para correção de possíveis erros.

A imagem obtida no pós-processamento está pronta para fornecer dados numéricos através da etapa de extração de dados, também conhecida como análise quantitativa. Algumas características dos objetos nas imagens são extraídas, resultando em algum tipo de discriminação entre as classes de objetos (AUGUSTO, 2012).

**Figura 8:** Fluxograma do Processamento e Análise de Imagem Digital.



Fonte: AUGUSTO, 2012

Dessa forma, segmentar uma imagem é fazer o agrupamento de pixels ou conjuntos de pixels de mesma propriedade. Ou seja, é fazer o desmembramento em regiões com propriedades comuns que correspondem a objetos ou partes de objetos contidos na mesma imagem ou que pertencem ao fundo (background) (CHAGAS, 2009).

As operações de segmentação e operações de extração de atributos são importantes e necessárias, pois com elas ocorrerá a isolação de regiões de pixels e uma série de parâmetros serão calculados, descrevendo o comportamento de tais regiões de interesse. A operação de segmentação mais comum é a limiarização. Um exemplo clássico é que tudo que estiver acima

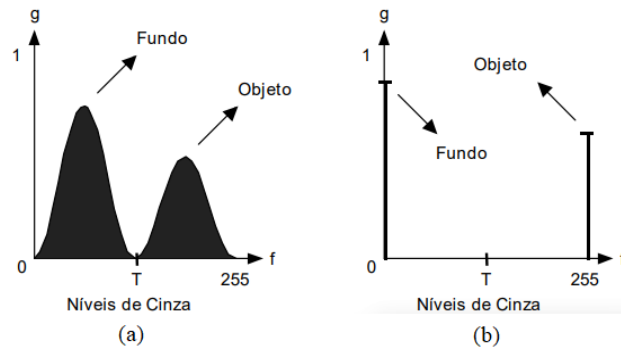
do corte, vira branco, e tudo que estiver abaixo vira preto, obtendo-se uma imagem binária (SCURI, 2002).

### 1.2.1 Limiarização (Thresholding)

O princípio da limiarização consiste em separar as regiões de uma imagem em duas classes: o fundo e o objeto. Este processo tem como efeito a criação de uma imagem binária na saída, devido a isso, essa técnica também é conhecida como binarização (MORGAN, 2008). Normalmente, o processo é realizado com a utilização de um histograma, ocorrendo a divisão do histograma em duas partes e convertendo os pixels cujo tom de cinza é maior ou igual a um determinado valor de limiar ( $T$ ) em brancos e os demais em pretos (FILHO, 1999).

Uma imagem com um fundo e um objeto bem distintos, em que os níveis de cinza apresentam uma variação claramente definida, resultará em um histograma com dois picos bem separados, e entre eles, uma região de valores relativamente baixos formando uma espécie de "vale". Na **Figura 9**, tem-se uma representação de um resultado de limiarização de uma imagem monocromática.

**Figura 9:** Limiarização de uma Imagem: (a) Histograma Original e (b) Histograma da Imagem Binarizada



Fonte: Filho, 1999.

Pode-se representar matematicamente a limiarização pela seguinte **Equação (2)**:

$$g(x, y) = \begin{cases} 1 & \text{se } f(x, y) > T \\ 0 & \text{se } f(x, y) \leq T \end{cases} \quad (2)$$

Sendo:

- $g(x, y)$ : imagem de saída após a limiarização;
- $f(x, y)$ : a imagem de entrada;
- $T$ : é o valor de limiar.

Se na **Equação 2**, a condição  $f(x, y) > T$  for verdadeira, o pixel representa o objeto e seu valor é definido como um (1). Caso contrário, o pixel representa o fundo e seu valor é definido como zero (0).

### 1.2.2 Modelo Paramétrico e não-Paramétrico

A detecção de objetos em uma imagem é uma tarefa fundamental. Existem diversos métodos para realizar essa tarefa. As técnicas baseadas em métodos estatísticos podem ser divididas em duas categorias diferentes: modelos paramétricos e não-paramétricos.

O método estatístico paramétrico considera que os dados normais são gerados a partir de uma distribuição específica (COSTA, 2014). Classificadores paramétricos tendem a uma menor probabilidade de erros de classificação. As funções de decisão deste tipo de classificador são ótimas por minimizarem a perda média de classificações erradas (AUGUSTO, 2012).

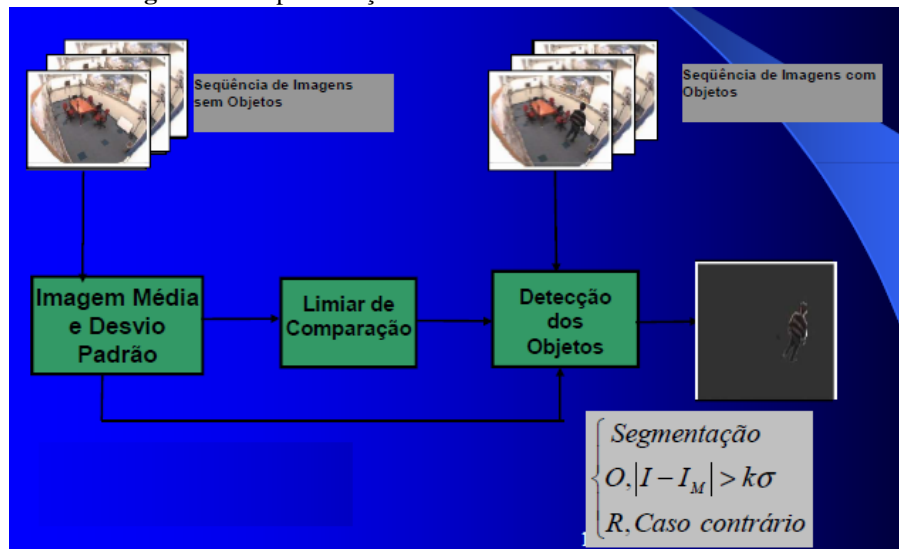
Nos modelos não-paramétricos, cada pixel no espaço de cores não tem uma distribuição conhecida e com isso é feita uma estimativa da distribuição de cada pixel para detectar um determinado objeto em uma imagem (OLIVEIRA, 2009). Por exemplo, métodos não paramétricos, a forma como os dados estão distribuídos não é definido inicialmente, mas determinado conforme os dados do conjunto de treinamento, fazendo com que sejam necessárias realizar menos suposições sobre os dados. Neste modelo é necessário garantir que a quantidade de dados de treinamento seja grande o suficiente para estimar bem a distribuição das amostras como um todo (COSTA, 2014).

### 1.2.3 Modelo Estatístico Paramétrico (Gaussiano)

O modelo da imagem de fundo ou referência é construído *pixel a pixel*. O valor de cada *pixel* observado é considerado resultante de um processo independente. O modelo estatístico paramétrico gaussiano para uma imagem de referência estática considera que cada pixel no espaço de cores tem uma distribuição normal. A imagem de referência é construída a partir do cálculo dos valores médios e dos desvios padrões de cada pixel. Esses cálculos são obtidos de um determinado número de sequências de imagens sem os objetos. Pode-se definir que os valores médios de cada pixel são utilizados como estimativas pontuais dos verdadeiros valores médios da população. E enquanto que os desvios padrões expressam a dispersão dos valores de cada pixel em função de ruídos e variações de iluminação no ambiente em que o sistema esteja inserido. Estas estimativas dependem do número (N) de quadros utilizados (OLIVEIRA, 2009).

Suponha-se que o espaço de cores em questão seja o YCbCr, na fase da segmentação, para cada quadro processado, o valor observado para cada *pixel* é comparado com a distribuição do seu correspondente na imagem de referência. A decisão para classificação de um *pixel* como pertencente à imagem de referência ou ao objeto é baseada nas informações de intensidade (Y) e de crominância (Cb, Cr) no espaço YCbCr. Para que um *pixel* seja classificado como pertencente à imagem de referência é necessário que os valores de duas componentes (Y e Cb) ou (Y e Cr) estejam dentro do desvio padrão dos valores correspondente na imagem de referência, caso contrário, será classificado como pertencente ao objeto (BOTINELLY, 2014). Na **Figura 10** tem-se uma representação deste método.

**Figura 10:** Representação do Modelo Estatístico Paramétrico.



Fonte: OLIVEIRA, 2009

### 1.2.3.1 Valor Médio

O valor médio é a média dos valores de N quadros obtidos para cada um dos *pixels* k (x, y) em cada componente de cor Y, Cb e Cr. Este cálculo é obtido pela **Equação 3**:

$$\bar{x}_k(Y, Cb, Cr) = \frac{1}{N} \sum_{i=0}^{N-1} I_{i(k)}(Y, Cb, Cr) \quad (3)$$

Sendo:

- i: número do quadro em processamento;
- k é a coordenada (x, y) de cada *pixel*;
- $I_{i(k)}(Y, Cb, Cr)$  é o valor corrente de cada *pixel* no espaço de cores YCbCr.

### 1.2.3.2 Desvio Padrão

O desvio padrão é comumente utilizado quando há variações na iluminação do ambiente e dos ruídos gerados pela câmera, ainda que a imagem seja estática, o mesmo *pixel* não terá valores idênticos ao longo do tempo. Levando em conta as variações de iluminação e do sensor da câmera, faz-se necessário computar as variações de cada *pixel*  $k$  em  $N$  quadros em cada componente de  $Y, Cb, Cr$ . Essas variações podem ser expressas pelo desvio padrão segundo a **Equação 4**:

$$S_k(Y, Cb, Cr) = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} [I_{i(k)}(Y, Cb, Cr) - \bar{x}_{i(k)}(Y, Cb, Cr)]^2} \quad (4)$$

### 1.2.3.3 Classificação dos pixels

Para classificar cada *pixel* como objeto ou pertencente à imagem de referência, a diferença absoluta (BD) de cada *pixel* é multiplicada pelo valor da função de compensação das variações do *pixel* (PCF) e o resultado é comparado com os valores dos limiares. As equações utilizadas para classificação de cada *pixel* estão reescritas a seguir:

$$F_k(Y, Cb, Cr) = |I_{(k)}(Y, Cb, Cr) - X_{(k)}(Y, Cb, Cr)| \quad (5)$$

$$Máscara \begin{cases} \frac{F_k(Y) > T_k(Y)}{ou} & \frac{F_k(Cb) > T_k(Y)}{ou} & \frac{F_k(Cr) > T_k(Cr)}{B, Se nenhuma das anteriores} \end{cases} \quad (6)$$

## 1.3 APLICAÇÃO CHROMA KEY

As aplicações de *chroma keying* têm se tornado cada vez mais difundidas em diversos campos, como cinema, ambiente virtual e realidade aumentada (CHU, 2017). A *Chroma Key* é considerada uma técnica robusta e importante para o processamento de imagem ou vídeo em que é amplamente utilizada. Uma clássica aplicação desta tecnologia é nas famosas previsões do tempo dos telejornais (TIWARI, 2014). Pode-se observar na **Figura 11** que o repórter do tempo fica à frente de uma tela verde. Enquanto isso, a cor de fundo verde é substituída por um mapa do tempo.

**Figura 11:** Aplicação de Chroma Key nas previsões do tempo nos Telejornais.

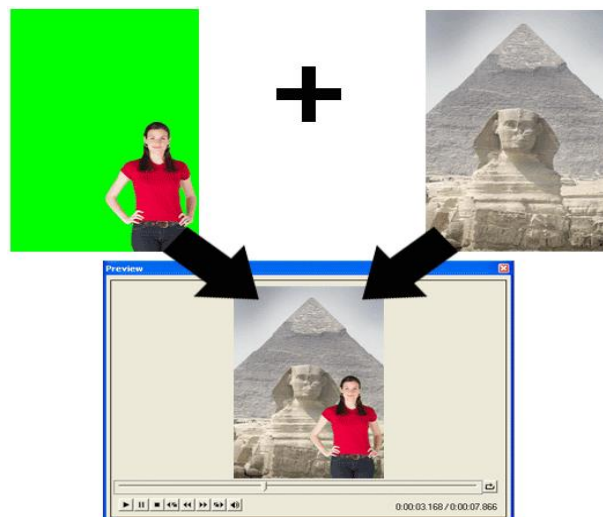


Fonte: KMS, 2014.

A tecnologia *Chroma key* é uma técnica, na qual um fundo verde ou azul fundo é colocado atrás do objeto de primeiro plano e, em seguida, faz-se a substituição desse fundo por outra uma imagem ou vídeo, criando-se um ambiente fictício. Essa técnica também é conhecida como codificação de cores ou sobreposição de separação de cores. Dessa forma, a tecnologia Chroma Key facilita a detecção de objetos em primeiro plano, garantindo assim uma precisão de informações que devem ser extraídas de uma determinada imagem com fundo verde ou azul (DIMITROPOULOS, 2010).

Resumidamente, esta técnica realiza a extração de objetos de um quadro de imagem e os combinam com um fundo qualquer a fim de compor novas cenas e criar efeitos especiais. (TIWARI, 2014). No quadro em que os objetos são segmentados é composto de duas partes que consistem em objetos de primeiro e segundo plano. Sendo os objetos de segundo plano uma cor sólida, normalmente verde ou azul. E os objetos de primeiro plano são os objetos colocados em frente desta tela verde ou azul, tal descrição pode ser observada na **Figura 12**.

**Figura 12:** Descrição do processo da técnica de *Chroma Key*.



Fonte: DESKSHARE.



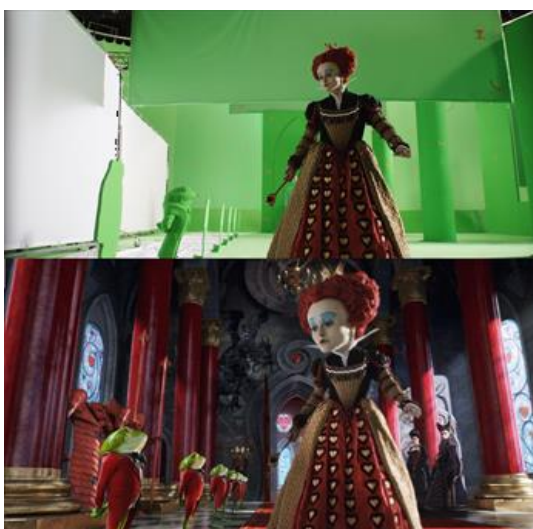
O uso da tecnologia Chroma Key pode garantir muitos benefícios. Tais como:

- a) A tecnologia Chroma Key aborda problemas de fundo fornecendo um fundo estável para todos os quadros da sequência;
- b) Facilita a detecção de objetos em primeiro plano, proporcionando mais precisão;
- c) Cenas diferentes podem ser sintetizadas usando a mesma sequência de quadros para objetos em primeiro plano e uma imagem de fundo diferente
- d) Finalmente, fornece robustez ao ruído da câmera, problemas de iluminação e possível desvio de cor entre câmeras.

Entretanto, esses fatores podem criar problemas significativos, especialmente, em grandes áreas de fundo com cor uniforme, afetando, assim, as áreas de pixel vizinhas pertencentes ao primeiro plano de objetos (DIMITROPOULOS, 2010).

Finalizando, o efeito de *Chroma-key* é muito utilizado em filmes, principalmente para criar efeitos especiais e/ou para economizar dinheiro, por exemplo, nas sessões de um filme que exigem locais estrangeiros ou qualquer ambiente diferente do normal, podem ser executados apenas dentro de um único estúdio (TIWARI, 2014). Pode-se observar na **Figura 13** um exemplo da aplicação desta técnica na indústria do cinema.

**Figura 13:** *Chroma Key* na indústria do Cinema.



Fonte: TRÊS, 2013.

## 2 METODOLOGIA

Neste capítulo serão abordados todos os aspectos metodológicos da pesquisa a ser desenvolvida, tendo como finalidade descrever todos os procedimentos necessários para desenvolver um sistema de detecção de objetos de uma imagem de referência estática utilizando o método estatístico paramétrico para aplicações em Chroma Key. Esse estudo teve por finalidade realizar uma pesquisa de natureza básica e aplicada.

Com o intuito de conhecer a problemática sobre a área de estudo e alcançar os objetivos almejados foi realizada uma pesquisa exploratória e bibliográfica

A seguir são listados os materiais utilizados ao longo do desenvolvimento da pesquisa e a ordem dos procedimentos metodológicos adotados.

### 2.1 AMBIENTE DE DESENVOLVIMENTO

O hardware utilizado nesta pesquisa tem a seguinte configuração: Processador Intel Core I7 (11ª geração - 2.8GHz, 12MB cache); RAM 8Gb; Placa de vídeo Intel Iris Xe Graphics; Sistema Operacional de 64bits, Windows 11 home.

### 2.2 MATERIAIS

Para o desenvolvimento, aquisição, testes e validação dos resultados obtidos, foram utilizados os seguintes materiais:

- Software MATLAB 2023a (MathWorks)
- 01 Tablet SAMSUNG S6 LITE;
- Editor de fotos Samsung;
- 01 Aparelho celular (Câmera);
- 01 Webcam embutida (notebook);
- 12 Folhas de papel emborrachado (40cm X 60cm);
- Aplicativo PhotoRoom;
- Aplicativo Picsart;
- Software Microsoft Excel 2019.

## 2.3 MÉTODOS

No diagrama de blocos da **Figura 14**, tem-se uma representação geral das etapas da metodologia utilizada no desenvolvimento deste projeto de pesquisa, quais sejam: Revisão bibliográfica; Definição do método de segmentação; Criação de imagens digitais (Testes Iniciais); Implementação do sistema em MATLAB (Validação Inicial); Métricas de avaliação; Criação do cenário de testes; Ajustes e definições dos parâmetros da câmera; Testes e avaliação de desempenho.

**Figura 14:** Etapas da Metodologia para o Desenvolvimento da Pesquisa.



Fonte: Autoria Própria.

No decorrer deste capítulo, será exposto de forma minuciosa e detalhada, a execução de cada etapa acima descrita no diagrama da **Figura 14**.

### 2.3.1 Revisão Bibliográfica

Nesta etapa do processo metodológico, foi realizado um levantamento bibliográfico e um estudo aprofundado sobre o estado da arte e das técnicas acerca do tema segmentação de imagens e aplicações em Chroma Key, objetivando selecionar ou definir alguma técnica de processamento de imagem que atendesse satisfatoriamente aos objetivos da pesquisa.

Os materiais de estudo foram obtidos principalmente das plataformas Google Academics e IEEE Xplore Digital Library, e esses materiais e demais documentos utilizados, encontram-se listados nas referências bibliográficas deste documento.

### 2.3.2 Definição do método de Segmentação e Criação de Imagens Digitais (Testes Iniciais)

De acordo com a temática proposta da pesquisa, nesta etapa da metodologia, foi definida qual a técnica de segmentação utilizada como objeto de estudo e análise dos resultados.

Inicialmente, para esta pesquisa, foram idealizadas duas técnicas de segmentação de

imagens que utilizavam como referências, modelos estatísticos paramétricos. Uma das técnicas é baseada no modelo que utiliza como parâmetro principal o cálculo das médias da imagem de fundo e a outra modelagem de segmentação utiliza como parâmetro principal o desvio padrão das imagens de referência de fundo.

Ainda nesta etapa metodológica, realizou-se a criação das imagens digitais, estas teriam como objetivo simular objetos frente a um fundo verde. Essa criação seria de crucial importância, pois essas imagens serão utilizadas para testes iniciais, com o objetivo de validar a performance do algoritmo desenvolvido em MATLAB. Para esse teste inicial, criou-se o quantitativo de imagens de acordo com a **TABELA 02** abaixo. Para a criação das imagens digitais, fez-se uso de um editor de fotos (Editor de Fotos Samsung - Tablet S6 Lite). O resultado dessas fotos será exposto no capítulo Implementação e Resultados.

**Tabela 02:** Quantitativo de imagens digitais (Testes Iniciais)

<b>Imagens</b>	<b>Quantidade</b>
Com Fundo Verde sem Objeto	30
Com Objetos Frente ao Fundo Verde	22

**Fonte:** Autoria própria.

### **2.3.3 Desenvolvimento em MATLAB (Validação Inicial)**

Após a definição dos métodos de segmentação de imagens a serem utilizadas na pesquisa e finalizada a criação das imagens digitais para os testes iniciais, iniciou-se o processo de implementação e desenvolvimento do algoritmo em MATLAB. Realizou-se a implementação de algoritmos para dois sistemas de cores: o RGB e o YCbCr. Ambos os sistemas de cores foram selecionados com objetivo de comparação, para verificar qual o melhor sistema se adequaria a temática proposta.

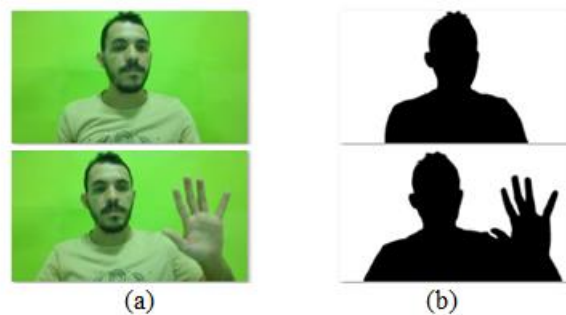
### **2.3.4 Métricas de Avaliação**

Finalizado o processo de implementação do algoritmo de segmentação em MATLAB e após os testes iniciais, implementou-se as métricas para avaliação do processo de segmentação, juntamente com o cálculo do fator de confiança ou erro. Esta etapa da metodologia tem por objetivo avaliar o método anteriormente implementado a fim de indicar a qualidade de desempenho do algoritmo.

A avaliação do método de detecção de objetos utiliza três métricas para quantificar o quanto cada quadro segmentado se aproxima da imagem de teste de referência (*ground truth*).

Durante as avaliações do método proposto, uma imagem de teste de referência foi construída para cada quadro das sequências de imagens utilizadas. Na **Figura 15** pode-se observar um exemplo de uma imagem de teste de referência. A imagem de teste de referência (**Figura 15-b**), os *pixels* do objeto são pretos e os da imagem de referência são todos brancos. Desta forma, para calcular o número de *pixels* do objeto é feita uma contagem dos pontos pretos. O resultado da contagem é comparado com o obtido pelo método de segmentação para avaliar o seu desempenho (OLIVEIRA, 2008).

**Figura 15:** Exemplo de sequências de imagens para avaliação do método de segmentação proposto – (a) imagem de teste; (b) imagem de teste de referência (ground truth).



Fonte: Autoria própria.

As métricas de avaliação são obtidas a partir das relações entre os parâmetros exibidos na **Tabela 1**:

**Tabela 1:** Parâmetros utilizados para obtenção das métricas de avaliação de métodos de detecção de objetos.

Detecção	Imagem de teste de referência ( <i>ground thuth</i> )	
	Positivo	Negativo
Positivo	TP	FP
Negativo	FN	TN

Fonte: OLIVEIRA, 2009.

Onde:

- TP - O verdadeiro positivo representa número de *pixels* pertencentes ao objeto corretamente detectados pelo método de segmentação.
- TN - Verdadeiro negativo representa o número de *pixels* pertencentes à imagem de fundo ou referência corretamente detectados pelo método de segmentação.
- FP- Falso positivo representa o número de *pixels* pertencentes à imagem de referência, erroneamente detectados como objetos.
- FN - Falso negativo representa o número de *pixels* do objeto erroneamente detectado como pertencente à imagem de referência.

Para indicar a qualidade do desempenho do algoritmo, foram usadas as seguintes métricas: Revocação (*Recall*), Precisão (*Precision*), Medida F (*F-measure*) e taxa de erro, todas expressas pelas equações abaixo:

$$R = \left( \frac{TP}{TP+FN} \right) \quad (7)$$

$$P = \left( \frac{TP}{TP+FP} \right) \quad (8)$$

$$F = \left( \frac{2.P.R}{P+R} \right) \quad (9)$$

$$Taxa\ de\ erro = \left( \frac{FN}{TP+FN} \right) \quad (10)$$

A revocação (R) avalia, para cada grupo resultante, quantos objetos pertencentes a ele são naturalmente do grupo sobre o total de objetos que deveriam fazer parte do grupo, dessa forma, é definida como a razão entre o número de *pixels* do objeto corretamente identificados pelo algoritmo ou método de segmentação e o número de *pixels* na imagem de teste de referência. Para se atingir um resultado de 100%, basta retornar apenas um grupo com todos os objetos existentes. A precisão (P) avalia quantos objetos de um grupo resultante do processo fazem parte da mesma classe sobre a quantidade total de objetos deste grupo, dessa forma, é definida como a razão entre o número de *pixels* corretamente identificados pelo algoritmo e o número de *pixels* detectados como objetos pelo algoritmo. E a medida F (F) combina as métricas de revocação e de precisão de forma a medir o quanto os agrupamentos possuem apenas os objetos quem fazem parte de uma mesma classe e todos os objetos desta classe, dessa forma, é uma média geométrica entre a revocação e a precisão. A taxa de erro (erro) é definida como a razão entre o número de *pixels* do objeto identificados erroneamente pelo algoritmo ou método de segmentação e o número de *pixels* na imagem de teste de referência (OLIVEIRA, 2009).

Para criação das imagens de referências de teste que seriam utilizadas nos cálculos das métricas, fez-se uso de dois aplicativos de edição de fotos (PhotoRoom e PicsArt).

### 2.3.5 Criação do Cenário de Testes e Ajustes da Câmera

Nessa etapa da metodologia, foi realizada a montagem do cenário de testes com objetos reais e pessoas. Inicialmente, usou-se 12 folhas de papel emborrachado (40cm X 60cm) para criar um grande painel verde retangular. A instalação deste painel ocorreu em um cômodo residencial, em que houvesse o maior controle da iluminação do ambiente a fim de evitar a criação de quaisquer sombras nesse painel, pois isso geraria ruídos e interferências no processo

da segmentação de imagens. Esta situação sobre as sombras será abordada em mais detalhes no capítulo Implementação e Resultados.

Ainda nesta etapa do processo, realizou-se ajustes em alguns parâmetros da câmera utilizada nesta pesquisa, com o objetivo de analisar se havia também interferências advindas da câmera durante o processo de captura das imagens e vídeos.

### **2.3.6 Testes e Avaliação de Desempenho**

Após a criação do cenário e ajustes da câmera, partiu-se para etapa de testes com objetos reais e pessoas. Os testes ocorreram utilizando-se os algoritmos que foram desenvolvidos nos espaços de cores RGB e YCbCr. Nesta etapa do processo metodológico, realizou-se testes com imagens estáticas reais e também testes de segmentação de vídeo em tempo real.

Além das métricas de avaliação supracitadas foi realizada uma avaliação final de desempenho do sistema desenvolvido em MATLAB. Essa avaliação final de desempenho fez-se uso de um método estatístico, denominado de inferência estatística T Student.

### 3 IMPLEMENTAÇÃO E ANÁLISE DOS RESULTADOS

Neste capítulo será abordado a respeito da implementação e análises dos resultados obtidos durante o desenvolvimento desta pesquisa. A temática deste projeto envolve uma técnica chamada *Chroma Key*, amplamente usada no mercado cinematográfico. Basicamente, este procedimento consiste em realizar a extração de objetos de interesse do quadro de imagem e combina-los com um fundo qualquer a fim de compor novas cenas, criando assim efeitos especiais.

Dessa maneira, o objetivo da pesquisa foi propor uma técnica de segmentação de imagem que atenda satisfatoriamente o resultado assim desejado quando comparado a tradicional técnica de Chroma Key, pois esta técnica utiliza um método denominado *Key Mean Clustering* que exige uma considerável quantidade de armazenamento por meio dos buffers que exigem maior velocidade e complexidade do sistema para implementação em hardware (TAI, 2014).

A partir de agora, será descrito em detalhes a implementação realizada, juntamente com as análises dos resultados na medida em que forem obtidos.

#### 3.1 DESENVOLVIMENTO DO ALGORITMO EM MATLAB E VALIDAÇÃO INICIAL

Como hipótese inicial para a pesquisa, selecionou-se duas técnicas de segmentação de imagens baseados em modelos estatísticos paramétricos. O primeiro método seria baseado na técnica média das imagens e o outro no desvio padrão. A priori, cada técnica seria implementada no sistema de cores RGB e YCbCr.

Para este primeiro momento, implementou-se o algoritmo em MATLAB (**APÊNDICE A**) em que utiliza como técnica de segmentação a média das imagens de fundo como referência para obtenção do objeto a ser detectado. Nesta etapa de validação do algoritmo, desenvolveu-se o script somente no sistema de cores RGB. O sistema de cor YCbCr será implementado e analisado na etapa final deste capítulo, em que ocorrerá a implementação das métricas e análise de desempenho dos algoritmos desenvolvidos.

Na **Figura 16**, tem-se um trecho do código fonte. Esta parte do algoritmo tem por objetivo realizar a leitura das imagens de fundo de referência, ou seja, leitura dos quadros de imagem que representa o fundo do cenário na cor verde sem a presença do objeto. Além dessa leitura, ocorre também o cálculo da média de cada elemento de imagem. Esse processo



representado na **Figura 16** consiste numa espécie de aprendizagem do algoritmo, pois futuramente o sistema entenderá o que é fundo de fato e o que é objeto a ser detectado.

**Figura 16:** Leitura das Imagens de Fundo de Referência e Cálculo da Média de Cada Elemento de Imagem.

```

N_FRAMES_BG = 30;
IMAGE_NAME=sprintf('fundo_verde-0');
%-----
% Calcula os valores médios de cada elemento de imagem
%-----
i_soma(1:RES_HOR,1:RES_VERT,1:3)=0;
for i=1:(N_FRAMES_BG),

    frame=imread(strcat(IMAGE_NAME,num2str(i),'.jpg'));
    i_atual=frame;

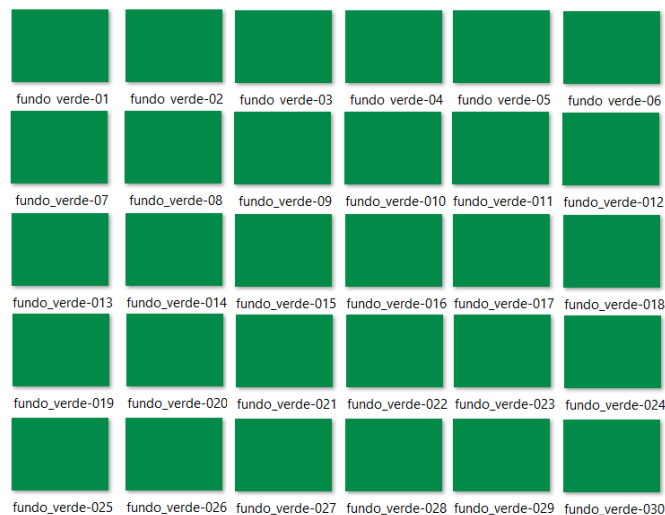
    i_atual=double(i_atual);
    i_soma = i_soma + i_atual;
end
i_media=(i_soma./N_FRAMES_BG );

```

**Fonte:** Autoria própria.

As imagens de fundo verde foram produzidas através de um editor de fotos do Tablet S6 Lite da Samsung. Essa produção digital de imagens serviu inicialmente para validação do algoritmo. Produziu-se 30 quadros de imagens com dimensões de 320x240, que representariam o fundo do cenário de cor verde, abaixo tem-se o resultado dessa produção na **Figura 17**:

**Figura 17:** Quadros de Imagens – Fundo Verde.



**Fonte:** Autoria própria.

Após realizada a leitura dos quadros de imagem de fundo, realizou-se a leitura das imagens com objetos frente a esse fundo verde. Pode-se observar uma parte do trecho do algoritmo que realiza essa função de aquisição/leitura na **Figura 18**:

**Figura 18:** Trecho Algoritmo - Leitura das Imagens de Testes

```

N_FRAMES_BG = 24;

for i=1:N_FRAMES_BG,

IMAGE_NAME=sprintf('f_verde_obje-0');
frame=imread(strcat(IMAGE_NAME,num2str(i),'.jpg'));

```

**Fonte:** Autoria própria.

Como já dito anteriormente, nesta etapa inicial de validação, fez-se a produção de imagens digitais para testes iniciais. Na **Figura 19**, tem-se a produção de 24 imagens de testes com objetos. Teve-se o cuidado de manter a mesma cor de fundo, pois caso ocorresse uma mudança na tonalidade isso prejudicaria o processo de segmentação do objeto.

**Figura 19:** Imagens de Testes com Objetos com Fundo Verde (Testes Iniciais).

**Fonte:** Autoria própria.

Realizada a leitura dos quadros de imagens do fundo e das imagens com o objeto, partiu-se para a implementação da segmentação que realizará a detecção do objeto em si.

Inicialmente, pré-determinou os parâmetros dos limiares RGB. Esses parâmetros servirão de apoio de comparação, ou seja, para determinar se um determinado elemento de imagem pertence ao fundo ou ao objeto. Além disso, teve-se a separação das componentes de cores em R, G e B. Essa separação ocorreu tanto para a imagem resultante da média (cálculo obtido da aquisição do fundo verde, anteriormente explícito), como para as imagens com o objeto. Na **Figura 20**, tem-se trechos do algoritmo que realiza tais procedimentos ditos anteriormente:

**Figura 20:** Definição dos Valores de Limiares e Separação das Componentes de Cores

```

%-----
LIMIAR_R = 80;
LIMIAR_G = 80;
LIMIAR_B = 80;
% Separação das componentes de cores R, G e B
im_media_r=i_media(:,:,1);
im_media_g=i_media(:,:,2);
im_media_b=i_media(:,:,3);
%-----
% Aquisição das imagens de teste
%-----
frame=imread(strcat(IMAGE_NAME,num2str(i),'.jpg'));
im_teste=double(frame);
im_teste_r=im_teste(:,:,1);
im_teste_g=im_teste(:,:,2);
im_teste_b=im_teste(:,:,3);

```

Fonte: Autoria própria.

Neste momento o algoritmo está pronto para realizar a segmentação e a detecção do objeto. Nesta etapa, o algoritmo realiza uma subtração entre a imagem corrente e a imagem média em cada componente de cor. Em seguida, ocorre uma localização e classificação dos elementos de imagem como pertencente ao objeto ou a imagem de fundo. Essa classificação ocorre a partir da comparação entre o elemento da imagem resultante da subtração com o limiar definido para cada componente de cor. Pode-se observar esses comandos na **Figura 21**.

**Figura 21:** Detecção do Objeto e Segmentação (Trechos do Algoritmo -APÊNDICE A)

```

%-----
% Cálculo da imagem diferença entre a imagem média e a imagem corrente
%-----
im_teste_rd=abs(im_teste_r - im_media_r);
im_teste_gd=abs(im_teste_g - im_media_g);
im_teste_bd=abs(im_teste_b - im_media_b);

%-----
% Classificação dos elementos de imagem como pertencente ao objeto (1) ou
% a imagem de fundo (0)
%-----
objeto1=find((im_teste_rd > LIMIAR_R)|...
            (im_teste_gd > LIMIAR_G)|...
            (im_teste_bd > LIMIAR_B));

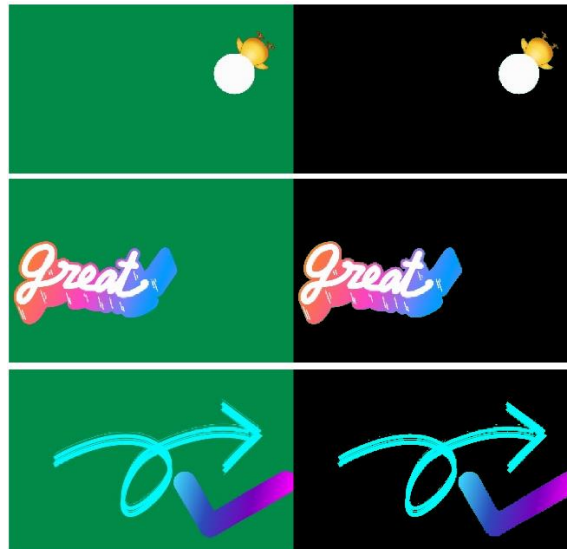
im_teste_f(1:RES_HOR,1:RES_VERT)=0;
im_teste_f(objeto1)=255;

```

Fonte: Autoria própria.

Pode-se observar na **Figura 22** os primeiros resultados obtidos dessa etapa de validação do algoritmo em RGB

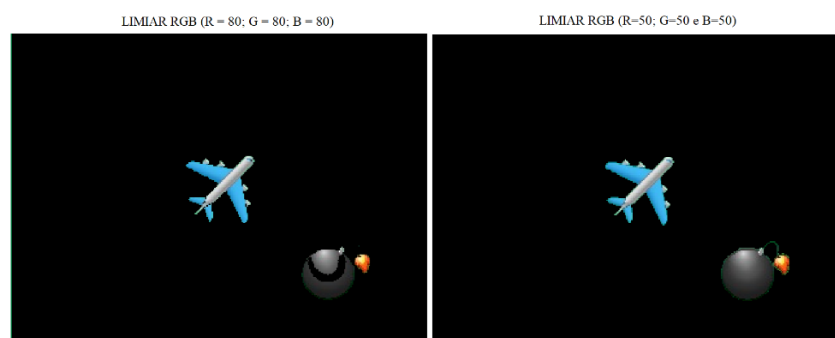
**Figura 22:** Resultado Segmentação RGB Limiar (R=50; G=50; B=50)



Fonte: Autoria própria.

Lembrando que este resultado foi obtido com os ajustes dos limiares RGB (R=50; G=50 e B=50). Limiares muito acima a esse valor ou muito abaixo, prejudicava e impactava no processo de seleção dos elementos de imagem durante o processo de segmentação. Pode-se observar melhor um exemplo disso na **Figura 23** para um ajuste de limiar em R=80, G=80 e B=80. Esse ajuste de limiar ocorreu de forma manual, ou seja, na tentativa e erro, sempre observando como o resultado final (objeto segmentado) respondia ao ajuste dos limiares, caso observasse uma segmentação não satisfatória, realizava o ajuste novamente do limiar RGB.

**Figura 23:** Comparação de Seleção de Diferentes Limiares RGB.



Fonte: Autoria própria.

De acordo com os resultados obtidos das segmentações baseado na média para o sistema de cor RGB, tiveram-se resultados satisfatórios, visto que os pixels pertencentes ao objeto foram classificados corretamente, sem haver a detecção de muitos falsos positivos

provenientes da imagem de referência de fundo. Dessa maneira, resolveu-se manter a linha de desenvolvimento da pesquisa apenas para esse parâmetro estatístico da média, excluindo-se o parâmetro desvio padrão conforme fora proposto no início da pesquisa.

Um ponto muito importante observado nesse início de implementação de validação dos algoritmos, foram em relação as alterações dos limiares. Identificou-se que tais mudanças não foram tão impactantes para o resultado final da segmentação das imagens de testes criadas através de editores de fotos.

Para as fotos criadas nesse primeiro momento de testes iniciais, a segmentação resultante não ficou muito comprometida quando comparadas com imagens obtidas de um cenário real. Esse motivo se deve ao fato de não haver nas fotos criadas por um editor, influências de iluminação, condições de captura da câmera, sombras, intensidade de brilhos e etc., pois esses fatores impactam no resultado final da segmentação.

Com isso, no próximo tópico, será abordado os resultados das segmentações de imagens obtidas a partir de um cenário real com fundo verde, tanto para imagens estáticas como processamento em vídeo.

### **3.2 CRIAÇÃO DO CENÁRIO CHROMA KEY E TESTES COM IMAGENS ESTÁTICAS REAIS.**

Conforme já dito no capítulo da metodologia, foi realizada a montagem do cenário de Chroma Key, em que se usou 12 folhas de papel emborrachado (40cm X 60cm) para criar um painel verde retangular. A instalação deste painel ocorreu em um cômodo residencial. Teve-se sempre a preocupação em relação a iluminação do ambiente, a fim de evitar a criação de quaisquer sombras nesse painel, pois isso geraria ruídos e interferências no processo da segmentação de imagens. Pode-se observar o resultado dessa criação do painel verde na **Figura 24**.

Após a criação do cenário, com o uso de um telefone celular com câmera, realizou-se a captura de algumas imagens para mais testes. Inicialmente, fez-se a captura somente do fundo verde sem objeto, em torno de 30 quadros de imagens.

**Figura 24:** Criação Cenário de Chroma Key



**Fonte:** Autoria própria.

Durante a aquisição das imagens, notou-se que a câmera estava realizando uma espécie de ajuste de foco e luminosidade automática, e isso seria um problema durante os testes, pois haveria mudanças nas tonalidades do fundo, fazendo com que o algoritmo durante a aprendizagem confundisse o fundo como se fosse um objeto. Na **Figura 25**, pode-se observar que mesmo sendo o mesmo fundo, houve uma grande diferença de mudança na tonalidade do fundo verde devido a essa estabilização automática da câmera.

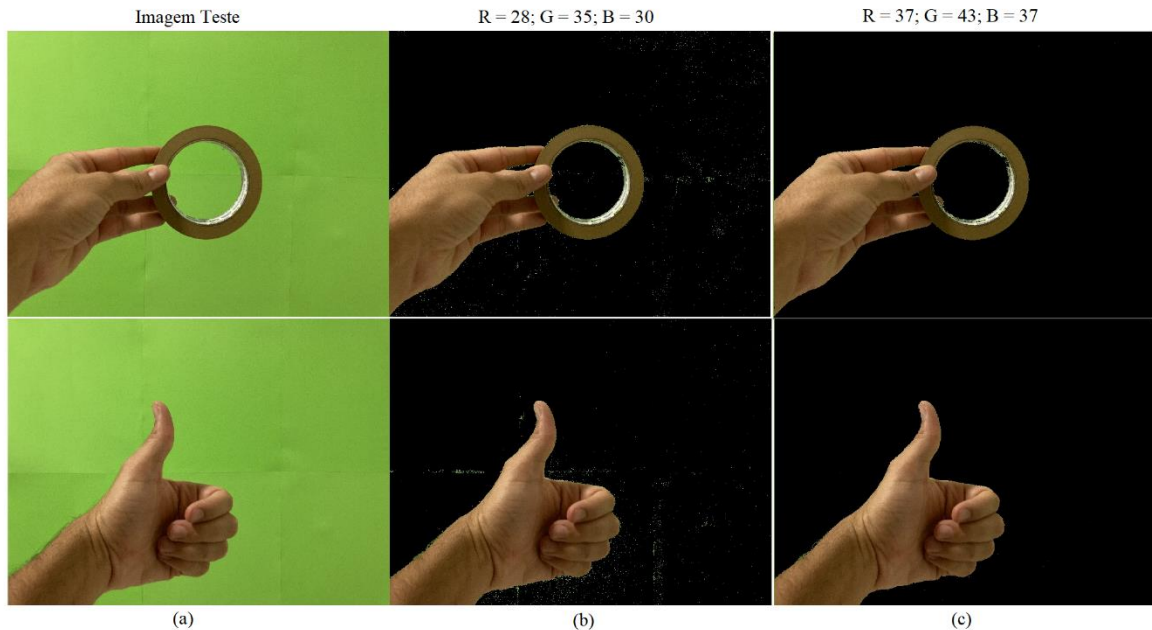
**Figura 25:** Ajuste de Foco e Luminosidade Automática.



**Fonte:** Autoria própria.

Após manter fixo os ajustes de foco e luminosidade da câmera celular, pode-se realizar os testes da segmentação. Como já dito anteriormente, nesta etapa de validação, os testes realizados apenas para o sistema de cores RGB. O resultado obtido encontra-se na **Figura 26**, abaixo.

**Figura 26:** Resultado Segmentação RGB Imagens Estáticas Reais com Diferentes Limiares.



Fonte: Autoria própria.

Observa-se que nesses testes com imagens reais, os ajustes de limiares tiveram impactos relevantes no resultado final da segmentação. Por exemplo, nas **Figuras 26-b** e **Figura 26-c**, os ajustes de limiares realizados para ambas as figuras não tem grande diferença numérica entre si, porém, no resultado final é nítida a quantidade ruídos que apareceram na **Figura 26-b**.

Dessa maneira, pode-se observar o impacto que a iluminação sob os objetos pode causar no resultado da segmentação.

O controle adequado da iluminação nos ambientes em que se encontra os objetos a serem segmentados é de extrema importância para garantir a qualidade e eficácia da técnica de segmentação utilizada, pois isso evita a criação de sombras indesejadas, mantém a consistência da cor de fundo conforme já dito anteriormente e reduz a criação de ruídos. A seguir, no próximo tópico, veremos a utilização da câmera para uma segmentação em tempo real e quais efeitos ocorreram durante o processo de implementação e testes.

### 3.3 SEGMENTAÇÃO EM TEMPO REAL COM E SEM AJUSTES DA CÂMERA DE VÍDEO

Nesta última etapa de validação do algoritmo, implementou-se o script para segmentação de imagens baseado no método estatístico paramétrico média em tempo real,

usando uma câmera de vídeo. O sistema de cor novamente utilizado para validação foi o RGB. O importante dessa etapa é a demonstração dos resultados que foram obtidos a partir das comparações realizadas do antes e após ao ajuste da câmera de vídeo.

Após a implementação do algoritmo (**Apêndice B**), iniciou-se os testes primeiramente sem o ajuste da câmera de vídeo. Pode-se observar o resultado obtido na **Figura 27**.

Verifica-se que há grandes influências, tanto dos objetos de refletirem luz, como da iluminação do ambiente sobre a câmera. Dessa maneira, notou-se que a câmera estava realizando ajustes automáticos e com isso, impactando no resultado final da segmentação. Por exemplo, ao analisar o resultado da segmentação em RGB da **Figura 27**, percebe-se que houve mudança na consistência da cor de fundo, seja clareando-o ou escurecendo-o quando na presença de objetos ou pessoas.

**Figura 27:** Resultados da Segmentação em RGB (Sem ajuste da Câmera de Vídeo).



**Fonte:** Autoria própria.

Diante dessas influências, necessitou-se realizar ajustes na câmera. Para isso, acessou-se as configurações da câmera pelo MATLAB para saber quais parâmetros seriam ajustados. Na **Figura 28**, tem-se as configurações da câmera nos padrões de fábrica. Após alguns estudos, análises e testes, foram selecionados e ajustados os parâmetros: Exposure, Saturation e o WhiteBalanceMode.



**Figura 28:** Parâmetros de Configurações da Câmera.

```

Name: 'Integrated Camera'
AvailableResolutions: {1x9 cell}
Resolution: '1280x720'
Exposure: -6
Brightness: 128
Gamma: 120
Sharpness: 3
Roll: 0
Saturation: 64
ExposureMode: 'auto'
WhiteBalance: 4600
BacklightCompensation: 1
Zoom: 100
WhiteBalanceMode: 'manual'
Tilt: 0
Pan: 0
ColorEnableMode: 'manual'
ColorEnable: 1
Contrast: 32
Hue: 0

```

**Fonte:** Autoria própria.

Na **Figura 29**, tem-se um trecho em MATLAB que realiza os ajustes de configurações da câmera toda vez que o algoritmo é executado, estes parâmetros foram alterados no algoritmo do **Apêndice B**.

**Figura 29:** Trecho Algoritmo - Ajustes e Configurações da Câmera

```

N_FRAMES_BG = 80;

for i=1:N_FRAMES_BG,

tic;
cam.Exposure = -5;
cam.WhiteBalanceMode = 'manual'
cam.Saturation = 80

```

**Fonte:** Autoria própria.

Após os devidos ajustes, obteve-se os resultados apresentados na **Figura 30**. Nota-se uma evidente melhoria no final da segmentação, ou seja, de fato estavam ocorrendo interferências em certos parâmetros na câmera e ocorrendo ajustes automáticos, prejudicando o resultado final.

**Figura 30:** Resultados da Segmentação em RGB (Com ajuste da Câmera de Vídeo).



Fonte: Autoria própria.

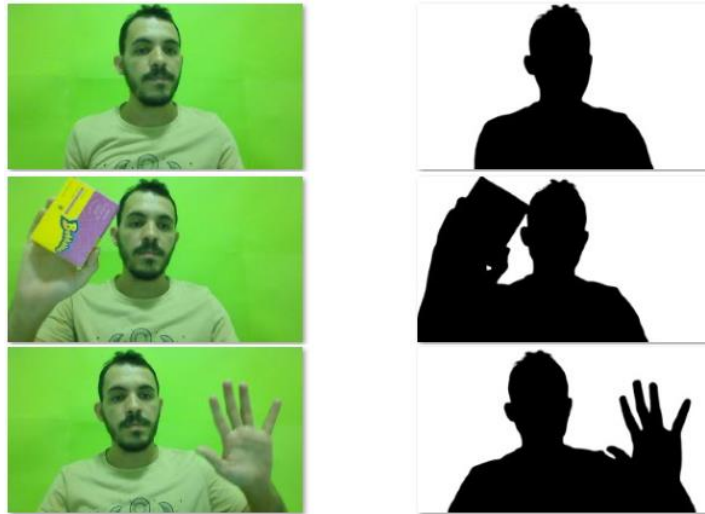
No próximo tópico, será demonstrado a implementação de métricas de avaliação para os algoritmos desenvolvidos. Essas métricas de avaliação irão mensurar a qualidade da segmentação desenvolvida e o desempenho dos algoritmos. Nesta fase será implementado para ambos os sistemas de cores a fim de concluir qual o algoritmo teve melhor resultado.

### 3.4 MÉTRICAS DE AVALIAÇÃO DE DESEMPENHO E INFERÊNCIAS ESTATÍSTICAS

Nesta etapa de desenvolvimento da pesquisa, implementou-se as métricas de avaliação para quantificar o desempenho dos algoritmos. Para avaliação do método de detecção de objetos foram utilizadas três métricas para quantificar o quanto cada quadro segmentado se aproximava da imagem de teste de referência.

Antes de implementar o cálculo das métricas em algoritmo, criou-se as imagens de referências das imagens que continham os objetos a serem detectados. Na **Figura 31**, pode-se observar o resultado dessa criação. Essas imagens de referência criadas servirão de apoio durante os cálculos das métricas, pois a imagem segmentada será comparada com essa imagem de referência, sendo assim demonstrando o quanto cada quadro segmentado se aproxima da imagem de teste de referência.

**Figura 31:** Imagens de Testes com suas Respectivas Referências.



**Fonte:** Autoria própria.

Na **Figura 32**, tem-se um trecho do algoritmo que realiza o cálculo dessas métricas. As métricas utilizadas como indicadores da qualidade de desempenho no algoritmo são denominadas revocação (*Recall*), precisão (*Precision*) e medida F (*F-measure*), e para verificar o número de acertos realizou-se o cálculo do fator de confiança ou erro.

**Figura 32:** Cálculo das métricas e do Fator de Confiança.

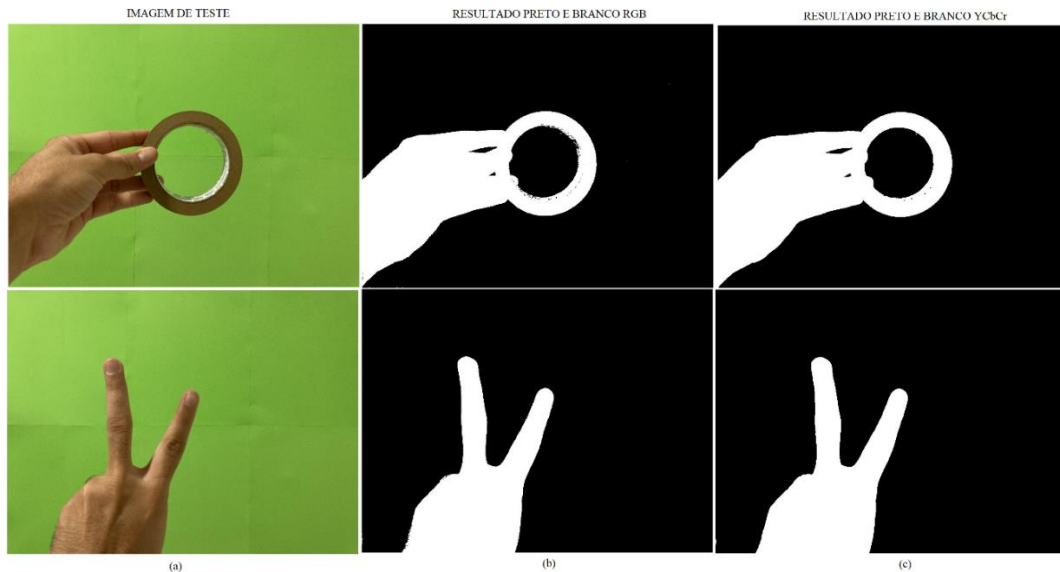
```
dr=(tp/(tp+fneg))*100; %Precision
r=(tp/(tp+fpos))*100; %revocação
f=((2*dr*r)/(dr+r)); %medida F
far=((fpos)/(tp+fpos))*100;

estatistica=[estatistica;numero_quadros,dr,r,f,far];
```

**Fonte:** Autoria própria.

Implementou-se dois algoritmos com métricas de avaliação para ambos os sistemas de cores (RGB e YCbCr). No **Apêndice C**, tem-se o algoritmo para o sistema de cores YCbCr com as métricas de avaliação implementadas. Primeiramente, pode-se observar na **Figura 33**, o resultado da segmentação em preto e branco para ambos os sistemas de cores.

**Figura 33:** Comparação dos Resultados da Segmentação para RGB e YCbCr.



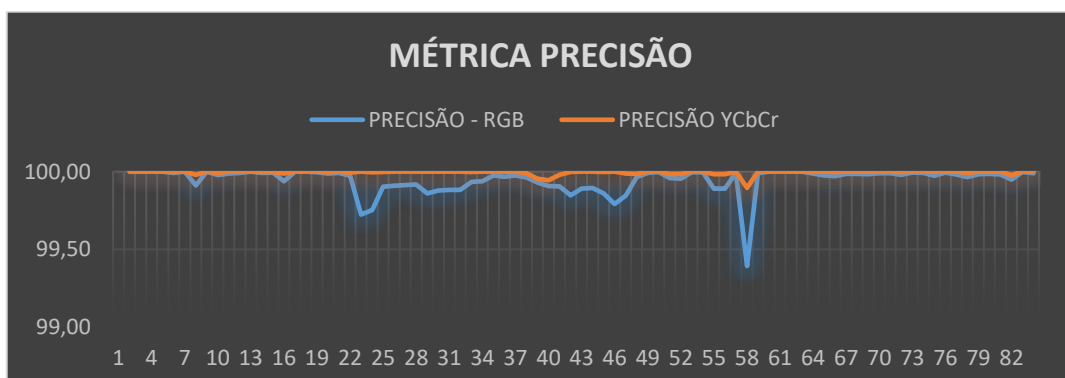
Fonte: Autoria própria.

Analisando-se as **Figuras 33-b** e **Figura 33-c**, nota-se que não há diferenças perceptíveis nos resultados obtidos para os dois sistemas de cores implementados. Entretanto, será feita uma inferência estatística a partir das métricas obtidas dos dois sistemas, para concluir se de fato algum dos dois sistemas de cores teve melhor performance ou independe do sistema de cor, pois o resultado será sempre satisfatório.

Dessa maneira, realizou-se primeiramente os cálculos das métricas de avaliação dos algoritmos nos sistemas de cores RGB e YCbCr. Pode-se observar graficamente os resultados para cada métrica de avaliação implementada nas **Figuras 34, 35 e 36**.

De acordo com os resultados obtidos, observa-se que nos dois planos de cores a precisão manteve-se entre 99 e 100%.

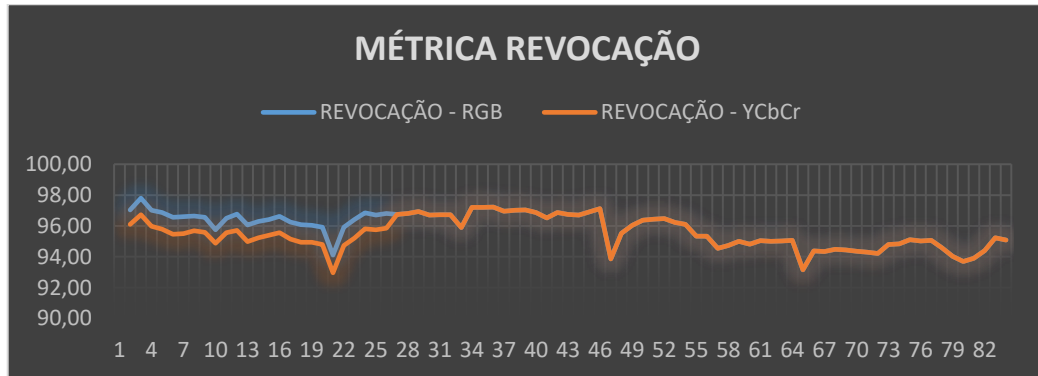
**Figura 34:** Gráfico de Desempenho dos Algoritmos nos Sistemas de Cores RGB e YCbCr – Métrica Precisão.



Fonte: Autoria própria.

Em relação a revocação, manteve-se entre 92,96 e 97,80% para ambos os sistemas de cores.

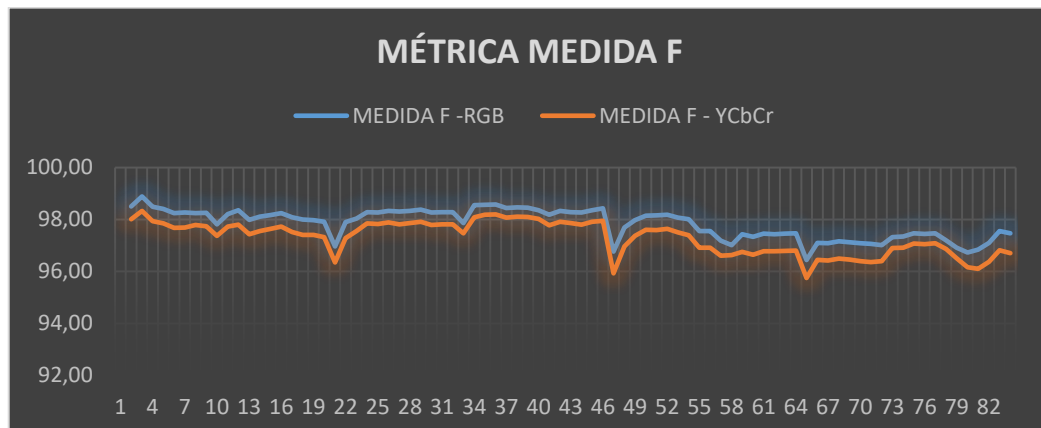
**Figura 35:** Gráfico de Desempenho dos Algoritmos nos Sistemas de Cores RGB e YCbCr – Métrica Revocação.



Fonte: Autoria própria.

Com relação a Medida F, observa-se que para os dois sistemas de cores a métrica manteve-se entre 95,75 e 98,89%.

**Figura 36:** Gráfico de Desempenho dos Algoritmos nos Sistemas de Cores RGB e YCbCr – Métrica Medida F.



Fonte: Autoria própria.

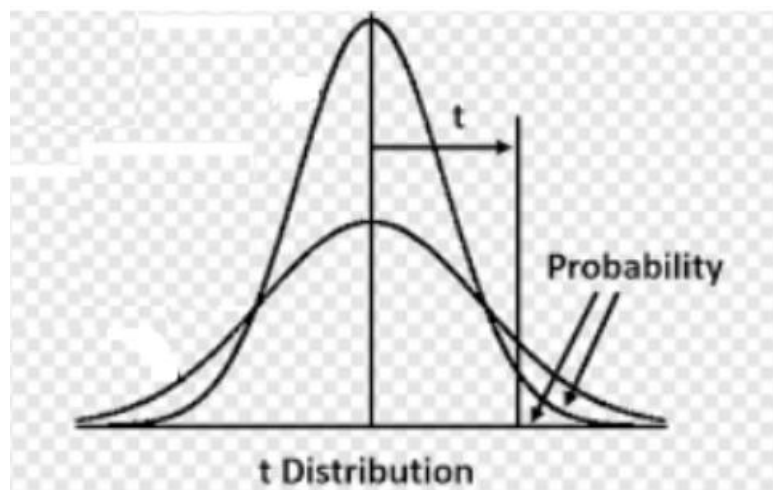
Após os cálculos das métricas de avaliação de ambos os sistemas de cores, partiu-se para avaliação das inferências estatísticas através da ferramenta de análise de dados do excel. Foi utilizado a função Teste-t: duas amostras presumindo variâncias equivalentes.

Na **Figura 37**, tem-se a etapa de preenchimento dos dados a serem analisados. No campo “Hipótese da Diferença de Média” preencheu-se com 0 (zero), isso significa que se considera que não há diferenças significativas no resultado da segmentação para ambos os sistemas de cores utilizados. Finalizado esta etapa de preenchimento dos dados para análise, obteve-se os resultados para cada métrica da avaliação e estão representados nas **Tabelas 04, 05, 06 e 07** localizadas no **Apêndice D**.

**Figura 37:** Preenchimento de Hipótese Inicial e Dados

**Fonte:** Autoria própria.

Ao analisar os dados das **Tabelas 04, 05, 06 e 07**, chegou-se à conclusão que de fato não há diferenças estatisticamente significativas nos resultados obtidos na segmentação, pois em todos os casos de análise para cada métrica, observou-se que o fator t-crítico se encontrava dentro do intervalo da Probabilidade ( $P[T \leq t]$ ), conforme pode-se observar a **Figura 38** que ilustra a análise de uma distribuição t student de forma geral. Dessa maneira, a partir das inferências estatísticas, ambos os sistemas de cores respondem satisfatoriamente aos resultados da segmentação obtidos.

**Figura 38:** Gráfico Distribuição t Student.

**Fonte:** Autoria própria.

Um ponto importante observado foi o tempo de processamento do sistema desenvolvido em MATLAB. Obteve-se um tempo de processamento de aproximadamente de 15s para um conjunto de 120 quadros processados.

Na figura 39 pode-se observar o resultado final obtido com aplicações em *Chroma Key*. Pode-se observar que teve uma leve criação de borda ao redor da imagem segmentada. Esse efeito é chamado de *spill* e ocorre normalmente devido ao objeto ficar muito próximo à tela verde, resultando em uma tonalidade esverdeada nas bordas do objeto. Este problema pode ser resolvido fazendo o distanciamento do objeto da tela.

**Figura 39:** Resultado Final da Segmentação com aplicação Chroma Key.



**Fonte:** Autoria própria.

## CONCLUSÃO

O processamento digital de imagens (PDI) desempenha um papel fundamental em diversas áreas da sociedade moderna. A capacidade de manipular, analisar e interpretar informações visuais de forma enriquecida e melhorada tem uma importância significativa em diversos setores do cotidiano, tais como: diagnósticos médicos; vigilância e segurança; entretenimento e mídia e entre outros. O PDI e o pré-processamento estão intimamente interligados. O pré-processamento é uma etapa crucial dentro do campo mais amplo do PDI e tem como objetivo preparar a imagem original para análises ou aplicações específicas, como segmentação, detecção de objetos e etc. Um exemplo dessa grande importância do pré-processamento é na utilização das aplicações em *Chroma Key*. Ao aprimorar a imagem original, eliminando ruídos e preparando o fundo cromático, aumenta a precisão da segmentação e permite que o objeto seja incorporado a um novo ambiente de maneira natural.

Para o presente projeto de pesquisa foi desenvolvido um sistema de detecção de objetos de uma imagem de referência estática utilizando o método estatístico paramétrico para aplicações em *Chroma Key*. O objetivo da pesquisa era desenvolver e validar um sistema de baixa complexidade na implementação quando comparado ao tradicional método de *Chroma Key*.

Inicialmente para pesquisa, foram idealizadas duas técnicas de segmentação a partir de modelos estatísticos paramétricos. Uma das técnicas seria baseada no modelo que utiliza como parâmetro principal o cálculo das médias da imagem de fundo e a outra modelagem de segmentação utilizaria como parâmetro principal o desvio padrão. No primeiro momento foi desenvolvido o sistema que utilizava como técnica de segmentação a média das imagens de fundo para a detecção do objeto. Diante dos resultados obtidos, percebeu-se a boa qualidade obtida na segmentação, e com isso, resolveu-se manter a linha de desenvolvimento da pesquisa apenas para esse parâmetro estatístico da média, excluindo-se o parâmetro desvio padrão conforme fora proposto no início da pesquisa.

Os primeiros resultados obtidos foram a partir de testes de imagens produzidas por um editor de fotos, apenas para validação inicial do algoritmo implementado. Viu-se que tais resultados foram satisfatórios. Dessa maneira, partiu-se então para a criação do fundo verde em um ambiente residencial e capturas de imagens reais para mais testes. Durante a aquisição



dessas imagens de testes, notou-se que a cor de fundo verde estava mudando sua tonalidade, devido a fatores externos como iluminação do ambiente e influências das reflexões de luz do objeto sobre a câmera. Com isso, necessitou realizar uma fixação do ajuste automático da câmera a fim de evitar que não ocorresse mais esses problemas de ajuste automática de brilho e luz do dispositivo.

O mesmo problema ocorreu durante os testes com o uso de uma câmera de vídeo, da mesma forma, teve-se que intervir diretamente nas configurações da câmera para evitar esses problemas. Ao analisar os resultados de antes e depois ao ajuste da câmera de vídeo, é perceptível como é crucial ter o controle não só do dispositivo da captura dos quadros de imagem, mas como também, é de extrema importância ter o controle adequado da iluminação do ambiente em que se encontra os objetos a serem segmentados, pois isso evita a criação de sombras indesejadas, mantém a consistência da cor de fundo e reduz a criação de ruídos.

Depois dos devidos ajustes para manter um bom resultado final na segmentação, teve-se a implementação das métricas de avaliação e as inferências estatísticas, que serviram de suporte para concluir qual o sistema de cor (RGB ou YCbCr) implementado no algoritmo para detecção de objetos obteve melhor desempenho. E observou-se que a partir das análises das inferências, ambos os sistemas de cores não possuíam diferenças significativas entre si, ou seja, tanto um quanto o outro demonstraram resultados satisfatórios.

Pode-se comprovar que houve um ótimo desempenho no sistema desenvolvido com baixa complexidade em sua implementação. De forma geral, a pesquisa agregou valor de conhecimento sobre a área do processamento digital de imagens, através de revisões bibliográficas, ampliando e enriquecendo assuntos em volta do tema que foi abordado neste projeto. A pesquisa possibilitará a exploração e o desenvolvimento para trabalhos futuros, como um possível estudo de implementação em hardware.

Acima de tudo, o trabalho possui grande importância para a comunidade acadêmica, pois poderá servir de apoio para futuras pesquisas nesta área, fornecendo grandes informações a respeito do tema desenvolvido.

## REFERÊNCIAS BIBLIOGRÁFICAS

ARAKI, J. **Processamento de Imagens**. [s.d.]. Apresentação do Power Point. Disponível em: <<https://www.pucsp.br/~jarakaki/cgpi/ProcessamentoImagensIntro.pdf>>. Acesso em 23 de jul. 2023.

AUGUSTO, K. **Identificação Automática do Grau de Maturação de Pelotas de Minério de Ferro**. Tese (Dissertação de Mestrado em Engenharia de Materiais e de Processos Químicos e Metalúrgicos) PUCRio. 2012. Disponível em:< [https://www.maxwell.vrac.puc-rio.br/21365/21365\\_1.PDF](https://www.maxwell.vrac.puc-rio.br/21365/21365_1.PDF)>. Acesso em: 20 de jun. de 2023.

BOTINELLY, V. M. F. **Arquitetura de Hardware em FPGA para Detecção de Objetos em Movimento Utilizando a Técnica de Subtração da Imagem de Fundo**. 2014. Monografia apresentada como pré-requisito para conclusão do Curso de Engenharia Elétrica, Escola Superior de Tecnologia, Universidade do Estado do Amazonas.

CHAGAS, E. M. P. F; RODRIGUES, D. L.; TAVARES, J. M. R. S. **Método de Segmentação de Objetos em Imagens Baseado em Contornos Activos e Algoritmo Genético**. 2009. Disponível em:< <https://web.fe.up.pt/~tavares/downloads/publications/artigos/METNUMp545.pdf>>. Acesso em: 02 de jul. 2023.

CHU, M.; ZHAO, J. **Chroma Keying Based on Stereo Images**. IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), 2017. Disponível em:<<https://ieeexplore.ieee.org/document/7986154>>. Acesso em: 22 de mai. 2023.

COSTA, G. **Detecção de anomalias utilizando métodos paramétricos e múltiplos classificadores**. Tese (Mestrado em Ciências de Computação e Matemática Computacional) - Instituto de Ciências Matemáticas e de Computação (USP). p. 14. 2014.

DESKSHARE. Disponível em: < [http://www.deskshare.com/Resources/articles/vem\\_green\\_screentechnique.aspx](http://www.deskshare.com/Resources/articles/vem_green_screentechnique.aspx)>, 2014. Acesso em: 23 maio 2023.

DIMITROPOULOS, K.; SEMERTZIDIS, T.; GRAMMALIDIS, N. **Improved 3d Video Synthesis Combining Graph Cuts and Chroma Key Technology**. 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video, 2010. Disponível em: <<https://ieeexplore.ieee.org/document/5506214>>. Acesso: 22 de mai. 2023.

FILHO, O. M.; NETO, H. V. **Processamento Digital de Imagens**. Rio de Janeiro: Brasport, 1999. Disponível em: <<https://www.ogemarques.com/wp-content/uploads/2014/11/pdi99.pdf>>. Acesso em: 19 de jul. de 2023.

KMS, Bobby Shaddox. iMovie Practice, Green Screens & Images/Graphics, 2014. Disponível em: <<http://mrshaddox212.blogspot.com.br/>>. Acesso em: 23 de mai. 2023.

LESTARI, P.; S, N.; KRISNANDI, D. **Depth Data based Chroma Keying using Grab-cut Segmentation**. International Conference on Computer, Control, Informatics and its Applications (IC3INA), 2018. Disponível em: <<https://ieeexplore.ieee.org/document/8629501>>. Acesso em: 20 de mai. 2023.

MARTINS, R. A.; SILVA, C. C. **As pesquisas de Newton sobre a luz: Uma visão histórica**. Conferência anual Internacional da Luz - Revista Brasileira de Ensino a Física 37, Oct - Dec 2015. Disponível em: <<https://doi.org/10.1590/S1806-11173731817>>. Acesso: 4 de jun. 2023.

MORGAN, J. **Técnicas de Segmentação de Imagens na Geração de Programas para Máquinas de Comando Numérico**. Tese de Dissertação de Mestrado em Engenharia da Produção. 2008. Disponível em: <<https://livros01.livrosgratis.com.br/cp109248.pdf>>. Acesso em: 12 de Jul. 2023.

NISHIDA, S. M.; OLIVEIRA, F. A. K.; TROLL, J. **Como vemos o mundo? As propriedades da luz**. [s.d.]. Disponível em: <[https://www2.ibb.unesp.br/Museu\\_Escola/2\\_qualidade\\_vida\\_humana/Museu2\\_qualidade\\_corpo\\_sensorial\\_visao3.htm](https://www2.ibb.unesp.br/Museu_Escola/2_qualidade_vida_humana/Museu2_qualidade_corpo_sensorial_visao3.htm)>. Acesso: 5 de jun. 2023.

OLIVEIRA, J. *Método para extração de objetos de uma imagem de referência estática com estimativa das variações de iluminação*. 2009. Tese (Doutorado em Engenharia Elétrica), Universidade Federal do Pará, Belém, 2009.

QUEIROZ, J. E. R; GOMES, H. M. **Introdução ao Processamento Digital de Imagens**. Revista RITA – Vol. VIII, N°. 2001. Disponível em: <[https://saulo.arisa.com.br/aulas/cg/PDI\\_RitaTutorial.pdf](https://saulo.arisa.com.br/aulas/cg/PDI_RitaTutorial.pdf)>. Acesso: 15 de jul. de 2023.

SCHADE, H. P.; LESTARI, P. **Boundary Matched Human Area Segmentation for Chroma Keying Using Hybrid Depth-Color Analysis**. IEEE 4th International Conference on Signal and Image Processing (ICSIP), 2019. Disponível em: <<https://ieeexplore.ieee.org/document/8868469>>. Acesso em: Acesso em: 20 de mai. 2023.

SCURI, A. E. **Fundamentos da Imagem Digital**. 2002. Disponível em: <<https://web.tecgraf.puc-rio.br/~scuri/download/fid.pdf>>. Acesso em: Acesso em 23 de jul. 2023.

SANTOS, T. S. **Segmentação Fuzzy de Texturas e Vídeos**. Dissertação de Mestrado em Sistemas e Computação - UFRN. 2012. Disponível em: <<https://repositorio.ufrn.br/bitstream/123456789/18063/1/TiagoSS DISSERT.pdf>>. Acesso em: 31 de jul. 2023.

SCURI, A. E. **Fundamentos da Imagem Digital**. Disponível em: <<https://web.tecgraf.puc-rio.br/~scuri/download/fid.pdf>>. Acesso em: 19 de jul. de 2023.

SILVA, S. S. **Dissertação de Mestrado: Segmentação de Imagens Utilizando Combinação de Modelos de Misturas Gaussianas -**. 2014. Disponível em: < <https://repositorio.ufpe.br/bitstream/123456789/11840/1/DISSERTA%c3%87%c3%83O%20Styve%20Stallone%20da%20Silva.pdf> >. Acesso em: 15 de mai. 2023.

TAI, N. N.; TRI, L. Q. B.; VINH, T. Q. **Chroma-Key Algorithm Based on Combination of K-Means and Confident Coefficients**. International Journal of Information and Electronics Engineering, Vol. 4, No. 3, May 2014. Disponível em:<<http://www.ijee.org/papers/432-F012.pdf>>. Acesso em: 20 de mai. 2023.

TIWARI, Anamika; BASKER, Geetha; KALAIARASI, P. **Fpga implementation for real time Chroma-Key effect using adaptive filter**. In: Proceedings of 4th IRF International Conference, Chennai, 9th March, 2014. Disponível em: <[https://www.digitalxplore.org/up\\_proc/pdf/50-1394697505168-172.pdf](https://www.digitalxplore.org/up_proc/pdf/50-1394697505168-172.pdf)>. Acesso: 22 de mai. 2023.

**TRÊS invenções que marcaram a história do cinema**, 2013. Disponível em: <https://oitavanocinema.wordpress.com/2013/03/14/tres-invencoes-que-marcaram-a-historiado-cinema/>>. Acesso em: 20 maio 2015.

## APÊNDICE A – ALGORITMO PARA VALIDAÇÃO INICIAL DO MÉTODO DE SEGMENTAÇÃO USANDO A MÉDIA.

```

%-----
%      UNIVERSIDADE DO ESTADO DO AMAZONAS
%-----
%      PROCESSAMENTO DIGITAL DE IMAGEM
%-----
% Orientando: Joel do Carmo Nogueira
%
% Orientador: Jozias Parente de Oliveira
%
% %-----
% Objetivo: Validação Inicial do Método de Segmentação Usando a Média
%-----
close all
clear

%-----
numero_quadros = 0;
tempo_processamento = 0;
%-----

N_FRAMES_BG = 30;
RES_HOR = 240;
RES_VERT = 320;

estatistica=[0,0,0,0,0];

%-----
LIMIAR_R = 50;
LIMIAR_G = 50;
LIMIAR_B = 50;

IMAGE_NAME=sprintf('fundo_verde-0');

%-----
% Calcula os valores médios de cada elemento de imagem
%-----
i_soma(1:RES_HOR,1:RES_VERT,1:3)=0;
for i=1:(N_FRAMES_BG),

    frame=imread(strcat(IMAGE_NAME,num2str(i),'.jpg'));
    i_atual=frame;

    i_atual=double(i_atual);
    i_soma = i_soma + i_atual;
end

i_media=(i_soma./N_FRAMES_BG );

%-----
% Separação das componentes de cores R, G e B

im_media_r=i_media(:,,1);
im_media_g=i_media(:,,2);
im_media_b=i_media(:,,3);
%-----

```

```

% Segmentação dos objetos das imagens de teste
%-----
N_FRAMES_BG = 22;

for i=1:N_FRAMES_BG,

tic;

IMAGE_NAME=sprintf('f_verde_obje-0');

%-----
% Aquisição das imagens de teste
%-----
frame=imread(strcat(IMAGE_NAME,num2str(i),'.jpg'));
im_teste=double(frame);
im_teste_r=im_teste(:,1);
im_teste_g=im_teste(:,2);
im_teste_b=im_teste(:,3);

%-----
% Cálculo da imagem diferença entre a imagem média e a imagem corrente
%-----
im_teste_rd=abs(im_teste_r - im_media_r);
im_teste_gd=abs(im_teste_g - im_media_g);
im_teste_bd=abs(im_teste_b - im_media_b);

%-----
% Classificação dos elementos de imagem como pertencente ao objeto (1) ou
% a imagem de fundo (0)
%-----
objeto1=find((im_teste_rd > LIMIAR_R)|...
            (im_teste_gd > LIMIAR_G)|...
            (im_teste_bd > LIMIAR_B));

im_teste_f(1:RES_HOR,1:RES_VERT)=0;
im_teste_f(objeto1)=255;

%-----
% Os elementos de imagem identificados como pertencentes
% ao objeto são substituídos pelos seus correspondentes
% valores da imagem de teste com o objeto
%-----
imfinal(1:RES_HOR,1:RES_VERT,1:3)=0;

imfinal_r(1:RES_HOR,1:RES_VERT)=0;
imfinal_g(1:RES_HOR,1:RES_VERT)=0;
imfinal_b(1:RES_HOR,1:RES_VERT)=0;

im_teste_r=uint8(im_teste_r);
im_teste_g=uint8(im_teste_g);
im_teste_b=uint8(im_teste_b);

imfinal_r(objeto1)=im_teste_r(objeto1);
imfinal_g(objeto1)=im_teste_g(objeto1);
imfinal_b(objeto1)=im_teste_b(objeto1);

imfinal(:,1)= imfinal_r;
imfinal(:,2)= imfinal_g;
imfinal(:,3)= imfinal_b;
%-----

```

```
% Exibição do video de entrada e o resultado da segmentação
imtestefinal=[frame,imfinal];

%-----
imtestefinal=imresize(imtestefinal,4,'nearest');
imtestefinal=uint8(imtestefinal);
imshow(imtestefinal);
pause (0.01);
%-----
numero_quadros = numero_quadros + 1;
tempo_processamento = tempo_processamento + toc;
quadros_por_segundo = numero_quadros/tempo_processamento;
%-----
%-----

end
```

## APÊNDICE B – ALGORITMO DE SEGMENTAÇÃO (MÉDIA) USANDO UMA CÂMERA DE VÍDEO

```

%-----
%      UNIVERSIDADE DO ESTADO DO AMAZONAS
%-----
%      PROCESSAMENTO DIGITAL DE IMAGEM
%-----
% Orientando: Joel do Carmo Nogueira
%
% Orientador: Jozias Parente de Oliveira
%
% %-----
% Objetivo: Validação do Método de Segmentação (Média). Usando Câmera de
% Vídeo
%-----

close all
clear

%-----
numero_quadros = 0;
tempo_processamento = 0;
%-----

N_FRAMES_BG = 40;
RES_HOR = 720;
RES_VERT = 1280;

%-----
LIMIAR_R = 45;
LIMIAR_G = 29;
LIMIAR_B = 37;

cam = webcam (1);

%-----
% Calcula os valores médios de cada elemento de imagem
%-----
i_soma(1:RES_HOR,1:RES_VERT,1:3)=0;
for i=1:(N_FRAMES_BG),

    cam.Exposure = -7; % Realiza o ajuste de luz na câmera
    cam.WhiteBalanceMode = 'auto'
    cam.Saturation = 64

    im_teste = snapshot(cam);
    i_atual=im_teste;
    i_atual=double(i_atual);
    i_soma = i_soma + i_atual;
end

i_media=(i_soma./N_FRAMES_BG );

%-----
% Separação das componentes de cores R, G e B

```



```

im_media_r=i_media(:,:,1);
im_media_g=i_media(:,:,2);
im_media_b=i_media(:,:,3);
%-----
% Segmentação dos objetos das imagens de teste
%-----
N_FRAMES_BG = 100;

for i=1:N_FRAMES_BG,

tic;
cam.Exposure = -7;
cam.WhiteBalanceMode = 'auto'
cam.Saturation = 64

frame = snapshot(cam);

%-----
% Aquisição das imagens de teste
%-----

im_teste=double(frame);

im_teste_r=im_teste(:,:,1);
im_teste_g=im_teste(:,:,2);
im_teste_b=im_teste(:,:,3);

%-----
% Cálculo da imagem diferença entre a imagem média e a imagem corrente
%-----
im_teste_rd=abs(im_teste_r - im_media_r);
im_teste_gd=abs(im_teste_g - im_media_g);
im_teste_bd=abs(im_teste_b - im_media_b);

%LIMIAR_R = (graythresh(im_teste_rd))*255;
%-----
% Classificação dos elementos de imagem como pertencente ao objeto (1) ou
% a imagem de fundo (0)
%-----
objeto1=find((im_teste_rd > LIMIAR_R)|...
            (im_teste_gd > LIMIAR_G)|...
            (im_teste_bd > LIMIAR_B));

im_teste_f(1:RES_HOR,1:RES_VERT)=0;
im_teste_f(objeto1)=255;

%-----
% Os elementos de imagem identificados como pertencentes
% ao objeto são substituídos pelos seus correspondentes
% valores da imagem de teste com o objeto
%-----
imfinal(1:RES_HOR,1:RES_VERT,1:3)=0;

imfinal_r(1:RES_HOR,1:RES_VERT)=0;
imfinal_g(1:RES_HOR,1:RES_VERT)=0;
imfinal_b(1:RES_HOR,1:RES_VERT)=0;

im_teste_r=uint8(im_teste_r);
im_teste_g=uint8(im_teste_g);

```

```
im_teste_b=uint8(im_teste_b);

imfinal_r(objeto1)=im_teste_r(objeto1);
imfinal_g(objeto1)=im_teste_g(objeto1);
imfinal_b(objeto1)=im_teste_b(objeto1);

imfinal(:,:,1)= imfinal_r;
imfinal(:,:,2)= imfinal_g;
imfinal(:,:,3)= imfinal_b;
%-----
% Exibição do video de entrada e o resultado da segmentação
imtestefinal=[frame,imfinal];

%-----
imtestefinal=imresize(imtestefinal,4,'nearest');
imtestefinal=uint8(imtestefinal);
imshow(imtestefinal);
pause (0.01);
%-----
numero_quadros = numero_quadros + 1;
tempo_processamento = tempo_processamento + toc;
quadros_por_segundo = numero_quadros/tempo_processamento;
%-----
%-----
end
```

## APÊNDICE C –SEGMENTAÇÃO DE IMAGEM NO SISTEMA DE COR YCBCR COM MÉTRICAS DE AVALIAÇÃO

```

%-----
%      UNIVERSIDADE DO ESTADO DO AMAZONAS
%-----
%      PROCESSAMENTO DIGITAL DE IMAGEM
%-----
% Orientando: Joel do Carmo Nogueira
%
% Orientador: Jozias Parente de Oliveira
%
% %-----
% Objetivo: Segmentação de Imagem (Parâmetro Média no Sistema de Cor YCbCr)
% com Métricas de Avaliação
% Vídeo

close all
clear

%-----
numero_quadros = 0;
tempo_processamento = 0;
%-----

N_FRAMES_BG = 37;
RES_HOR = 720;
RES_VERT = 1280;

estatistica=[0,0,0,0,0];

%-----
LIMIAR_y = 25;
LIMIAR_cb = 15;
LIMIAR_cr = 15;

IMAGE_NAME=sprintf('tst-');

%-----
% Calcula os valores médios de cada elemento de imagem
%-----
i_soma(1:RES_HOR,1:RES_VERT,1:3)=0;
for i=1:(N_FRAMES_BG),

    frame=imread(strcat(IMAGE_NAME,num2str(i),'.jpg'));
    i_atual=frame;

    i_atual=double(i_atual);
    i_soma = i_soma + i_atual;
end

i_media=(i_soma./N_FRAMES_BG );

%Conversão RGB para YCbCr

i_media=uint8(i_media);
i_media=rgb2ycbcr(i_media);

```

```

%-----

%-----
% Separação das componentes de cores Y, Cb e Cr

im_media_y=i_media(:,1);
im_media_cb=i_media(:,2);
im_media_cr=i_media(:,3);

im_media_y=double(im_media_y);
im_media_cb=double(im_media_cb);
im_media_cr=double(im_media_cr);

%-----
% Segmentação dos objetos das imagens de teste
%-----
N_FRAMES_BG = 120;

for i=38:N_FRAMES_BG,

tic;

IMAGE_NAME=sprintf('tst-');

%-----
% Aquisição das imagens de teste
%-----
frame=imread(strcat(IMAGE_NAME,num2str(i),'.jpg'));

% Conversão RGB da Imagem Teste para YCbCr
%
% frame=uint8(frame);
frame=rgb2ycbcr(frame);

im_teste=double(frame);

im_teste_y=im_teste(:,1);
im_teste_cb=im_teste(:,2);
im_teste_cr=im_teste(:,3);

%-----
% Cálculo da imagem diferença entre a imagem média e a imagem corrente
%-----
im_teste_yd=abs(im_teste_y - im_media_y);
im_teste_cbd=abs(im_teste_cb - im_media_cb);
im_teste_crd=abs(im_teste_cr - im_media_cr);

% LIMIAR_R = (graythresh(im_teste_rd))*255;
%-----
% Classificação dos elementos de imagem como pertencente ao objeto (1) ou
% a imagem de fundo (0)
%-----
objeto1=find((im_teste_yd > LIMIAR_y)|...
            (im_teste_cbd > LIMIAR_cb)|...
            (im_teste_crd > LIMIAR_cr));

im_teste_f(1:RES_HOR,1:RES_VERT)=0;

```

```

im_teste_f(objeto1)=255;
%imtestefinal=im_teste_f;
%-----
% Identifica os elementos de imagem pertencentes ao objeto
% em cada componente de cor individualmente
%-----
%-----
% Os elementos de imagem identificados como pertencentes
% ao objeto são substituídos pelos seus correspondentes
% valores da imagem de teste com o objeto
%-----
imfinal(1:RES_HOR,1:RES_VERT,1:3)=0;
imfinal=uint8(imfinal);

imfinal_y(1:RES_HOR,1:RES_VERT)=0;
imfinal_cb(1:RES_HOR,1:RES_VERT)=0;
imfinal_cr(1:RES_HOR,1:RES_VERT)=0;

im_teste_y=uint8(im_teste_y);
im_teste_cb=uint8(im_teste_cb);
im_teste_cr=uint8(im_teste_cr);

imfinal_y(objeto1)=im_teste_y(objeto1);
imfinal_cb(objeto1)=im_teste_cb(objeto1);
imfinal_cr(objeto1)=im_teste_cr(objeto1);

imfinal(:,,1)= imfinal_y;
imfinal(:,,2)= imfinal_cb;
imfinal(:,,3)= imfinal_cr;

%-----
% Exibição do vídeo de entrada e o resultado da segmentação
% frame=ybcr2rgb(frame);
imfinal=ybcr2rgb(imfinal);
imtestefinal=[frame,imfinal];

%-----
%imtestefinal=imresize(imtestefinal,4,'nearest');
%imtestefinal=uint8(imtestefinal);
imshow(imtestefinal);
pause (0.01);
%-----
numero_quadros = numero_quadros + 1;
tempo_processamento = tempo_processamento + toc;
quadros_por_segundo = numero_quadros/tempo_processamento;
%-----
%-----

%-----
% Cálculo das métricas de avaliação dos resultados
%-----

IMAGE_REF=sprintf('tst-o-');

%-----
% Aquisição da imagem de referência para
% cálculo estatístico
%-----
im_ref=imread(strcat(IMAGE_REF,num2str(i),'.jpg'));
im_ref=double(im_ref);

```

```

im_ref=fix(im_ref*255);
im_ref=im_ref(:,:,1);
%-----
%-----
% Determinação do número de pixels do objeto e da imagem de fundo
%-----

fgref=find(im_ref==0); % Localiza o objeto na imagem de referência
[size_fgref,n]=size(fgref);

if(size_fgref==0)
    size_fgref = 1;
end

bgref=find(im_ref>0);
[size_bgref,n]=size(bgref);

%----- Estatística Objeto Final -----
fgobj=find( (im_teste_f(fgref))>0);
[size_fgobj,n]=size(fgobj);

if(size_fgobj==0)
    size_fgobj = 1;
end

bgobj=find( (im_teste_f(bgref))==0);
[size_bgobj,n]=size(bgobj);

extracted_foreground_pixels=size_fgobj;
reference_foreground_pixels=size_fgref;
extracted_background_pixels=size_bgobj;
reference_background_pixels=size_bgref;

tp=extracted_foreground_pixels;

if(size_bgobj<size_bgref)

    fpos=size_bgref-size_bgobj;
    pos_error=(fpos/size_bgref)*100;

else

    fpos=0;
    pos_error=0;

end

if(size_fgobj<size_fgref)

    fneg=size_fgref-size_fgobj;
    neg_error=(fneg/size_fgref)*100;

else

    fneg=0;
    neg_error=0;

```

```
end

dr=(tp/(tp+fneg))*100; %Precision
r=(tp/(tp+fpos))*100; %revocação
f=((2*dr*r)/(dr+r)); %medida F
far=((fpos)/(tp+fpos))*100;

estatistica=[estatistica;numero_quadros,dr,r,f,far];

%xlswrite ('teste3',estatistica)

End
```

## APÊNDICE D – TABELAS DAS INFERÊNCIAS ESTATÍSTICAS (DISTRIBUIÇÃO T STUDENT).

**Tabela 4** – Inferência Estatística da Métrica Precisão

	<i>PRECISÃO - RGB</i>	<i>PRECISÃO YCBCR</i>
<b>MÉDIA</b>	99,94723958	99,99472879
<b>VARIÂNCIA</b>	0,007184093	0,000207331
<b>OBSERVAÇÕES</b>	83	83
<b>VARIÂNCIA AGRUPADA</b>	0,003695712	
<b>HIPÓTESE DA DIFERENÇA DE MÉDIA</b>	0	
<b>GL</b>	164	
<b>STAT T</b>	-5,032339463	
<b>P(T&lt;=T) UNI-CAUDAL</b>	6,31466E-07	
<b>T CRÍTICO UNI-CAUDAL</b>	1,654197929	
<b>P(T&lt;=T) BI-CAUDAL</b>	1,26293E-06	
<b>T CRÍTICO BI-CAUDAL</b>	1,974534576	

**Fonte:** Autoria própria.

**Tabela 05:** Inferência Estatística Métrica Revocação

	<i>REVOCAÇÃO - RGB</i>	<i>REVOCAÇÃO - YCBCR</i>
<b>MÉDIA</b>	95,82953033	95,51253439
<b>VARIÂNCIA</b>	1,176892437	1,04236402
<b>OBSERVAÇÕES</b>	83	83
<b>VARIÂNCIA AGRUPADA</b>	1,109628229	
<b>HIPÓTESE DA DIFERENÇA DE MÉDIA</b>	0	
<b>GL</b>	164	
<b>STAT T</b>	1,938603534	
<b>P(T&lt;=T) UNI-CAUDAL</b>	0,027133293	
<b>T CRÍTICO UNI-CAUDAL</b>	1,654197929	
<b>P(T&lt;=T) BI-CAUDAL</b>	0,054266587	
<b>T CRÍTICO BI-CAUDAL</b>	1,974534576	

**Fonte:** Autoria própria.



Tabela 06: Inferência Estatística da Métrica F

	<b>MEDIDA F -RGB</b>	<b>MEDIDA F - YCBCR</b>
<b>MÉDIA</b>	97,84189651	97,30381876
<b>VARIÂNCIA</b>	0,316773756	0,398163242
<b>OBSERVAÇÕES</b>	83	83
<b>VARIÂNCIA AGRUPADA</b>	0,357468499	
<b>HIPÓTESE DA DIFERENÇA DE MÉDIA</b>	0	
<b>GL</b>	164	
<b>STAT T</b>	5,797625981	
<b>P(T&lt;=T) UNI-CAUDAL</b>	1,68249E-08	
<b>T CRÍTICO UNI-CAUDAL</b>	1,654197929	
<b>P(T&lt;=T) BI-CAUDAL</b>	3,36498E-08	
<b>T CRÍTICO BI-CAUDAL</b>	1,974534576	

Fonte: Autoria própria.

Tabela 07: Inferência Estatística do Fator de Confiança

	<b>FATOR DE CONFIANÇA - RGB</b>	<b>FATOR DE CONFIANÇA - YCBCR</b>
<b>MÉDIA</b>	4,170469669	4,487465609
<b>VARIÂNCIA</b>	1,176892437	1,04236402
<b>OBSERVAÇÕES</b>	83	83
<b>VARIÂNCIA AGRUPADA</b>	1,109628229	
<b>HIPÓTESE DA DIFERENÇA DE MÉDIA</b>	0	
<b>GL</b>	164	
<b>STAT T</b>	-1,938603534	
<b>P(T&lt;=T) UNI-CAUDAL</b>	0,027133293	
<b>T CRÍTICO UNI-CAUDAL</b>	1,654197929	
<b>P(T&lt;=T) BI-CAUDAL</b>	0,054266587	
<b>T CRÍTICO BI-CAUDAL</b>	1,974534576	

Fonte: Autoria própria.