



Detecção da Maturidade de Tomates



UNIVERSIDADE FEDERAL DO CEARÁ

JULHO DE 2025



**UNIVERSIDADE
FEDERAL DO CEARÁ**



UNIVERSIDADE
FEDERAL DO CEARÁ

Curso: Engenharia de Computação
Disciplina: Seminários de Computação I
Professor: Marcelo Marques Simões de Souza
Semestre: 2025.1

Equipe:

Cizé Lucas Gome Lima - 538864
Pedro Henrique Moraes da Silva - 521461
Danilo Victor Pacheco De Lima - 555394



UNIVERSIDADE
FEDERAL DO CEARÁ

ARTIGO PRINCIPAL

Computers and Electronics in Agriculture 146 (2018) 43–50



Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag



Original papers

A methodology for fresh tomato maturity detection using computer vision

Peng Wan^a, Arash Toudeshki^b, Hequn Tan^a, Reza Ehsani^{b,*}

^a College of Engineering, Huazhong Agricultural University, No. 1, Shizishan Street, Wuhan 430070, China

^b School of Engineering, University of California, Merced, CA 95343, USA



ARTICLE INFO

Keywords:

Maturity detection

Feature color value

Backpropagation neural network

Tomato

2010 MSC:

00-01

99-00

ABSTRACT

Recent advancements in computer vision have provided opportunities for new applications in agriculture. Accurate yield estimation of fruit and vegetable crops is very important for better harvesting and marketing planning and logistics. This paper proposes a method for detecting the maturity levels (green, orange, and red) of fresh market tomatoes (Roma and Pear varieties) by combining the feature color value with the backpropagation neural network (BPNN) classification technique. A maturity detection device based on computer vision technology was designed specifically to acquire the tomato images in the lab. The tomato images were processed and the targets of the tomatoes were obtained based on the image processing technology. After that, the maximum inscribed circle of the tomato's surface was identified as the color feature extraction area. The color feature extraction area was divided into five concentric circles (sub-domains). The average hue values of each sub-region were extracted as the feature color values and used to describe the maturity level of the samples. After that, the five feature color values were imported to the BPNN as input values to detect the maturity of the tomato samples. Analysis of the results shows that the average accuracy for detecting the three maturity levels of tomato samples using this method is 99.31%; and the standard deviation is 1.2%.

Fonte:

<https://www.sciencedirect.com/science/article/abs/pii/S016816991730056X>



UNIVERSIDADE
FEDERAL DO CEARÁ

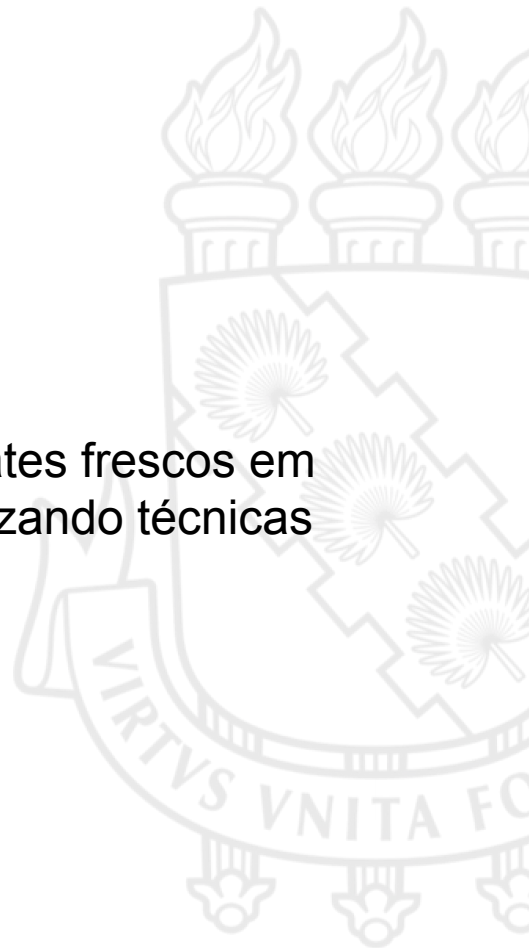
ROTEIRO

- 1** **Parte 1:**
- Introdução
- 2** **Parte 2:**
- Metodologia
- 3** **Parte 3:**
- Implementação
- 4** **Parte 4:**
- Resultados
- 4** **Parte 5:**
- Conclusão



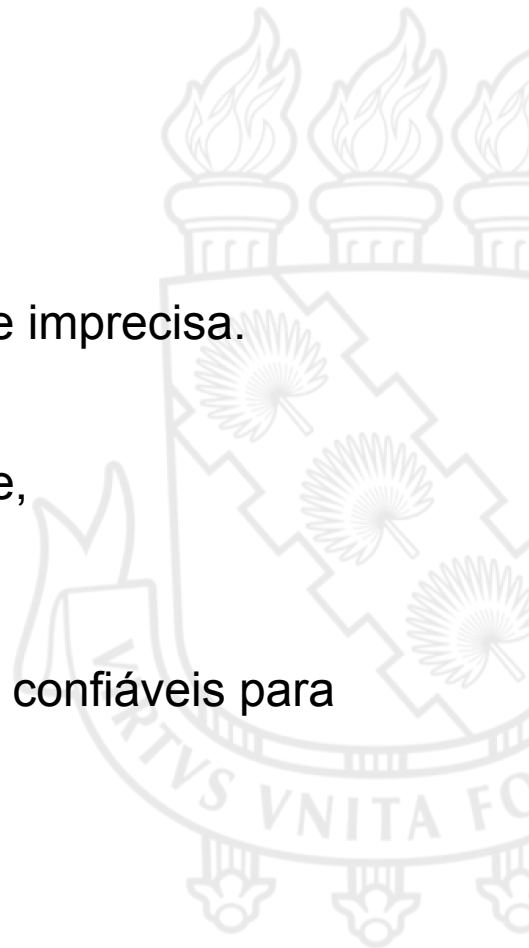
INTRODUÇÃO

O Artigo apresenta uma metodologia para classificar tomates frescos em três estágios de maturação (verde, laranja, vermelho) utilizando técnicas de visão computacional e redes neurais.



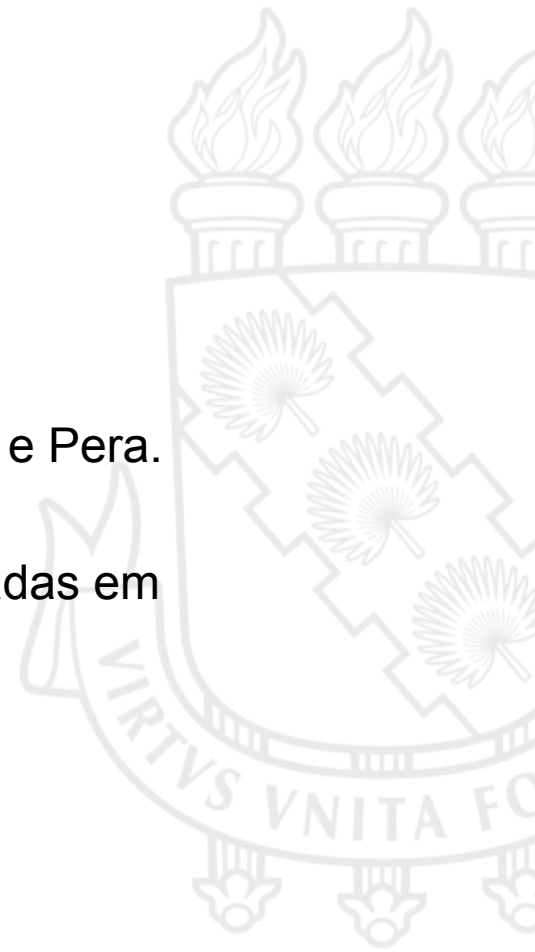
Introdução - Motivação

- A classificação manual de tomates é lenta, subjetiva e imprecisa.
- O estágio de maturação influencia colheita, transporte, armazenamento e preço.
- Há necessidade de métodos rápidos, padronizados e confiáveis para avaliação em larga escala.



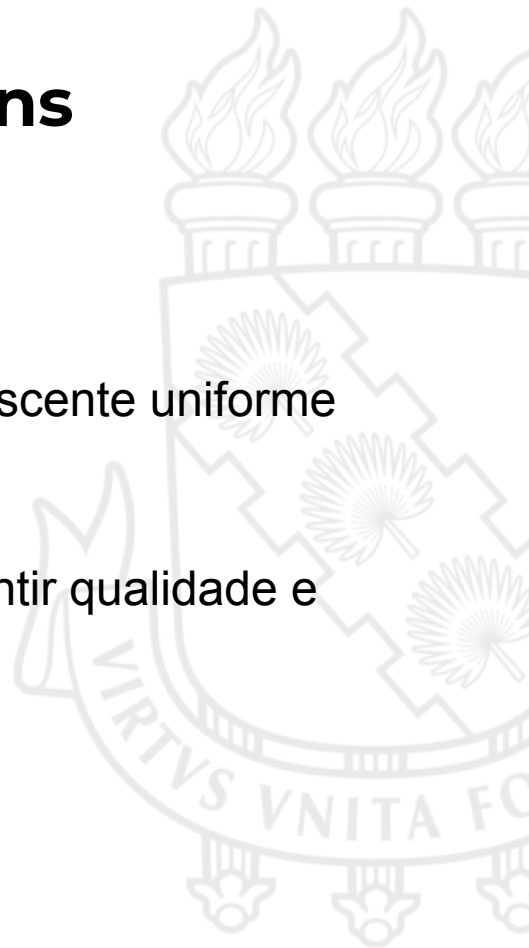
METODOLOGIA - Coleta de Amostras

- Foram utilizadas duas variedades de tomate: Roma e Pera.
- Cerca de 150 amostras por variedade foram analisadas em laboratório.



METODOLOGIA - Aquisição de imagens

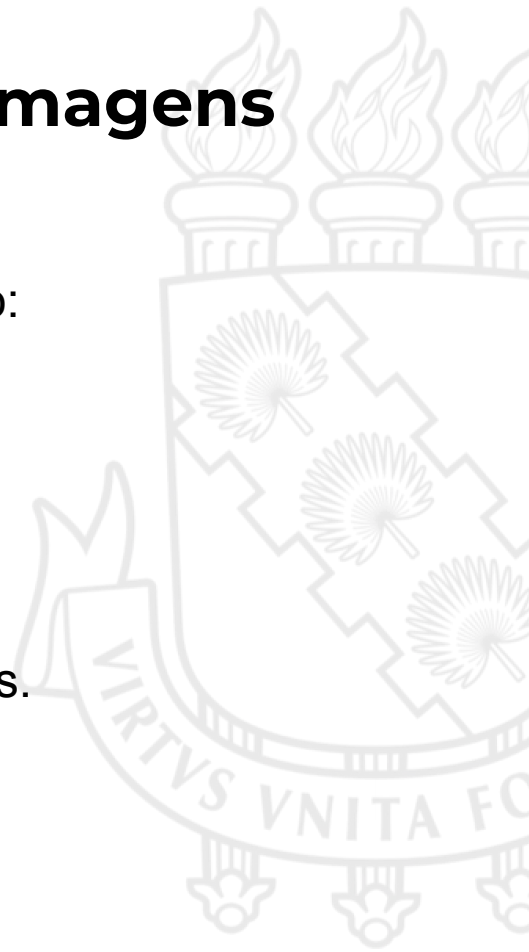
- As imagens foram capturadas com iluminação fluorescente uniforme e fundo preto.
- Foi utilizado um sistema de câmera digital para garantir qualidade e repetibilidade.



METODOLOGIA - Processamento de imagens

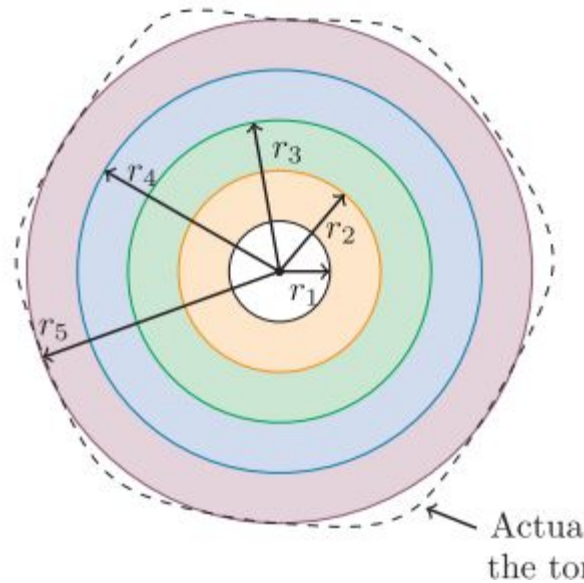
As imagens passaram por técnicas de pré-processamento:

- Segmentação por limiar (thresholding)
- Redução de ruído
- Extração de contornos e preenchimento de bordas.



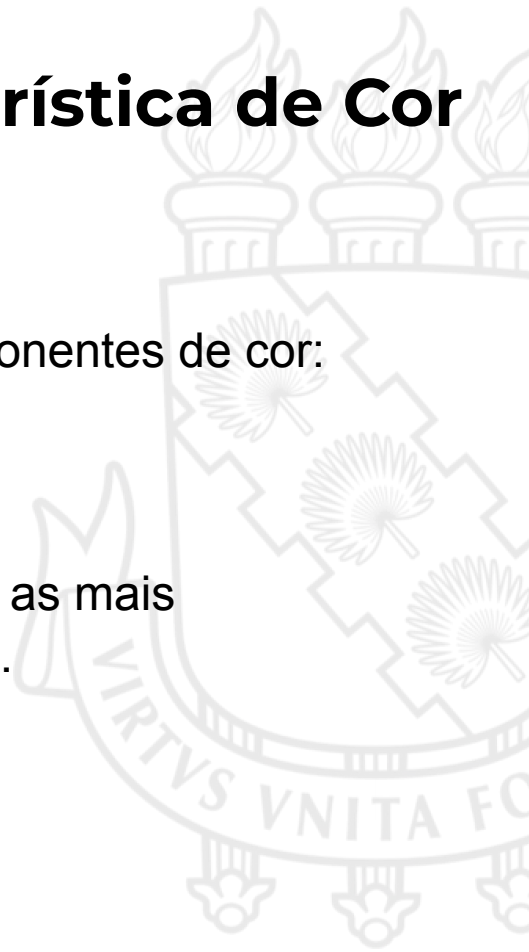
METODOLOGIA - Extração de Característica de Cor

- A área de interesse do tomate foi identificada por um círculo inscrito máximo.
- A área foi dividida em subáreas FCA1 á FCA5.

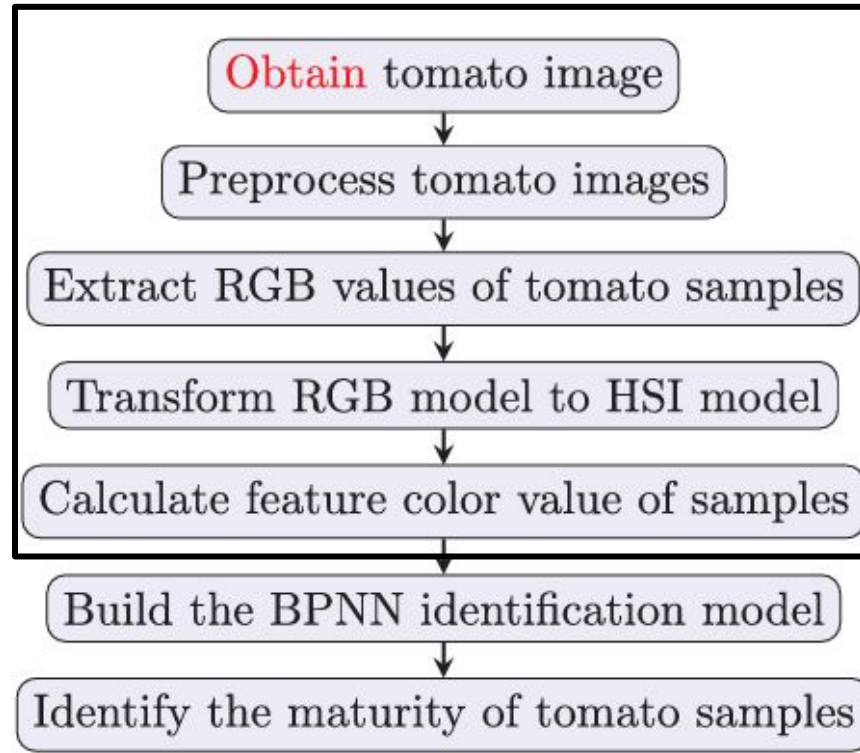


METODOLOGIA - Extração de Característica de Cor

- Em cada região foram extraídas médias das componentes de cor:
 - RGB (vermelho, verde, azul)
 - HSI (matiz, saturação, intensidade)
- As componentes Hue (H) e Blue (B) mostraram-se as mais relevantes para diferenciar os níveis de maturação.



SEGMENTAÇÃO DOS TOMATES



Fonte:
Figura 03 do Artigo principal



Segmentação de tomates - O problema dos Datasets



Fonte: Dataset 03

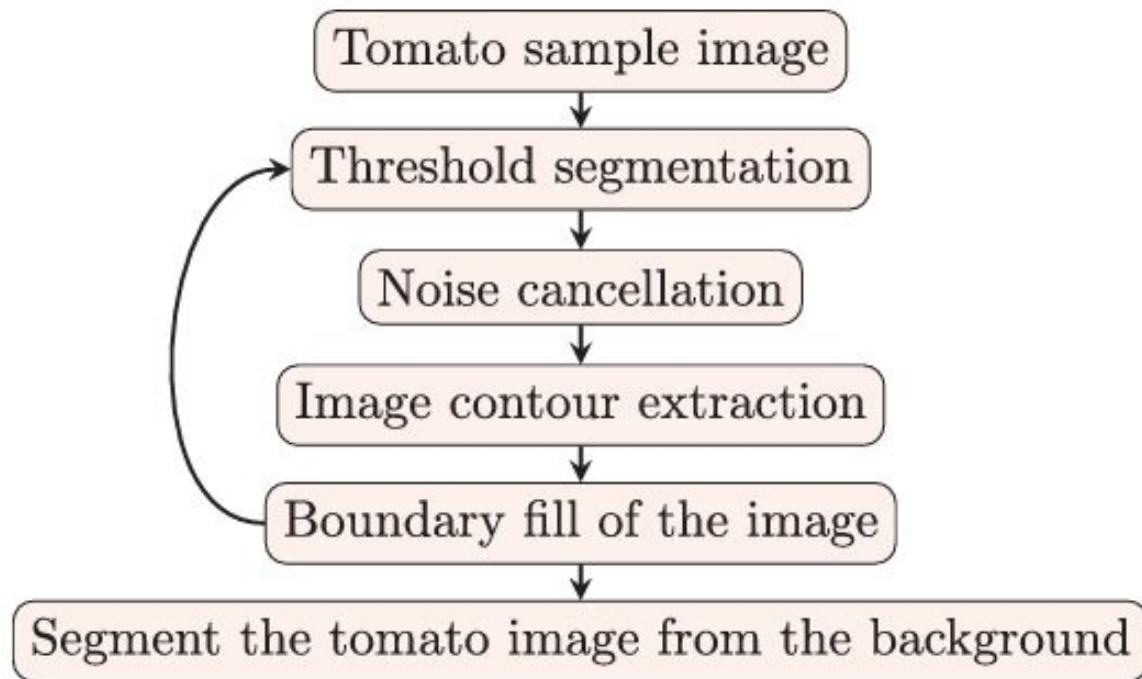


Fonte: Dataset 02



Fonte: Dataset 01

Segmentação de tomates - Etapas da Segmentação



Fonte:
Figura 04 do Artigo principal

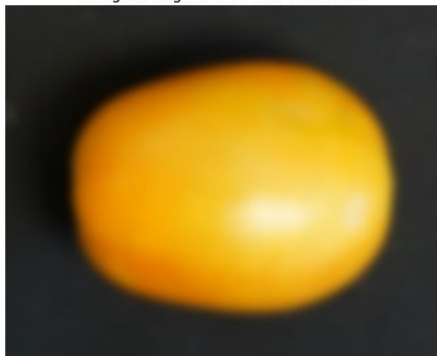


Segmentação de tomates - Thresholding Global Único

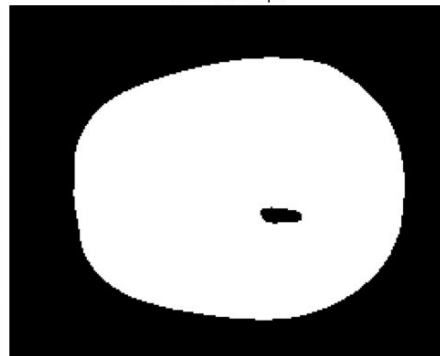
Imagem Original



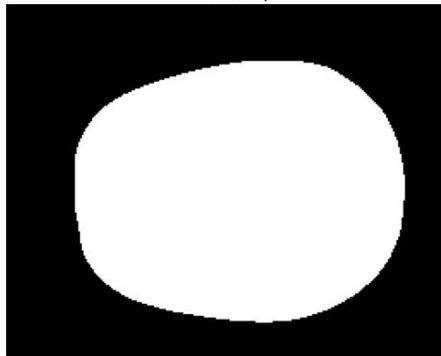
Imagem Original com Blur Gaussiano



Máscara Limpa



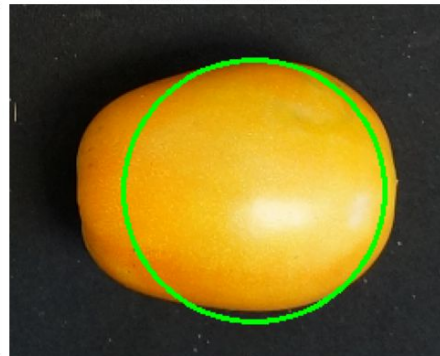
Máscara Limpa



Máscara Aplicada



Círculo Inscrito no Tomate



Segmentação de tomates - Thresholding Global Único

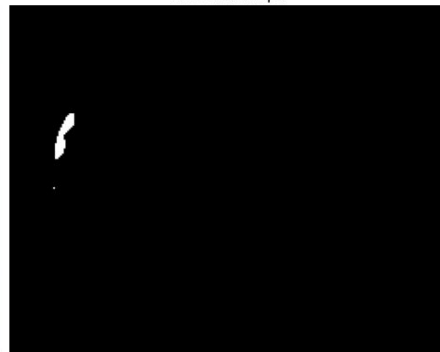
Imagem Original



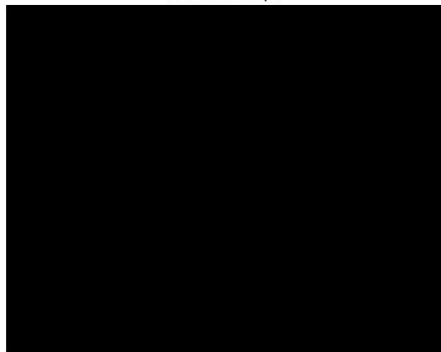
Imagem Original com Blur Gaussiano



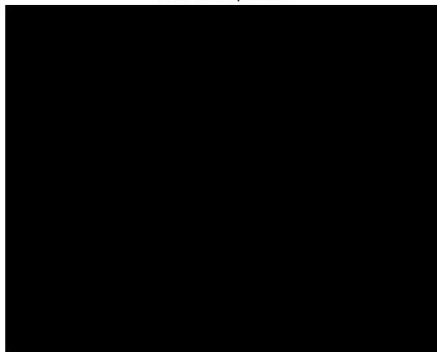
Máscara Limpa



Máscara Limpa



Mascara Aplicada



Círculo Inscrito no Tomate



Segmentação de tomates - Thresholding Global Múltiplo

Imagem Original

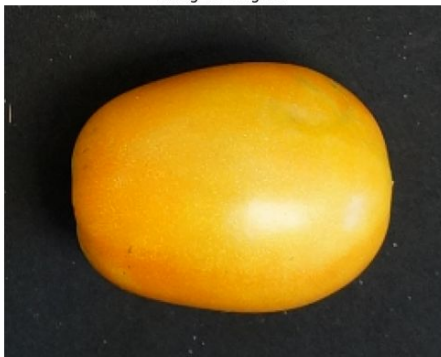
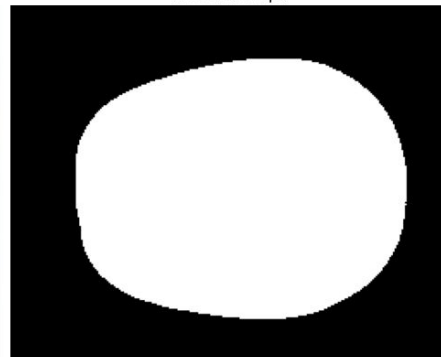


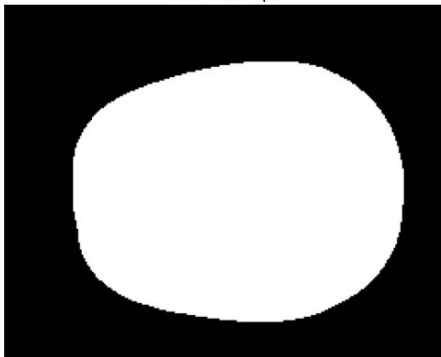
Imagem Original com Blur Gaussiano



Máscara Limpa



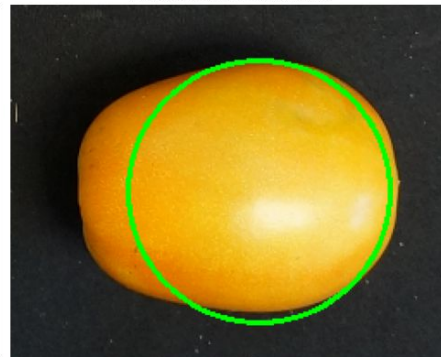
Máscara Limpa



Máscara Aplicada



Círculo Inscrito no Tomate



Segmentação de tomates - Thresholding Global Múltiplo

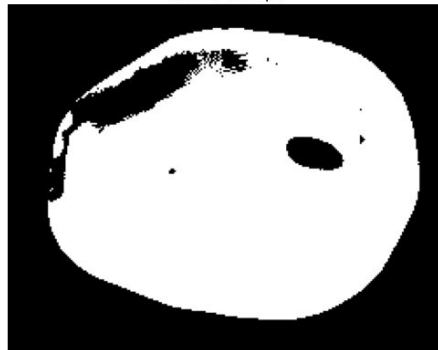
Imagem Original



Imagem Original com Blur Gaussiano



Máscara Limpa



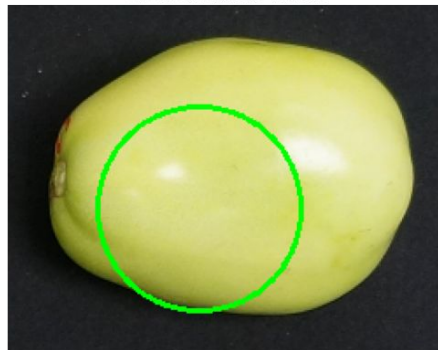
Máscara Limpa



Mascara Aplicada



Círculo Inscrito no Tomate

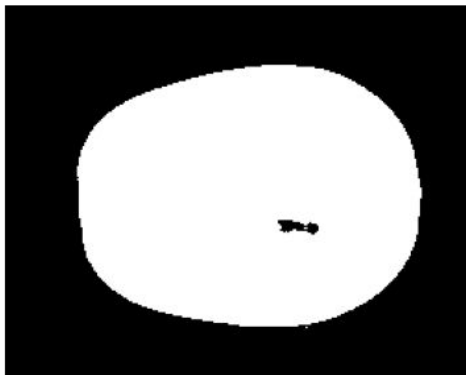


Segmentação de tomates - Solução: Thresholding Adaptativo

Imagem Original RGB



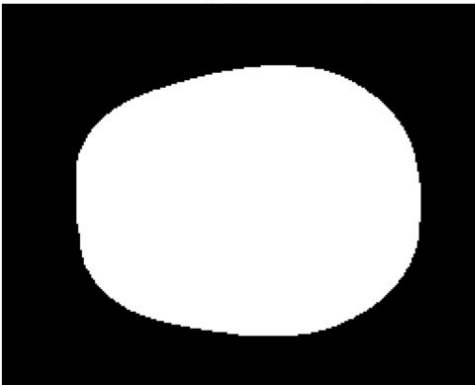
Máscara obtida



Máscara Aplicada



Máscara Limpa (Antes do Re-Fill)



Máscara Re-Preenchida (refill)

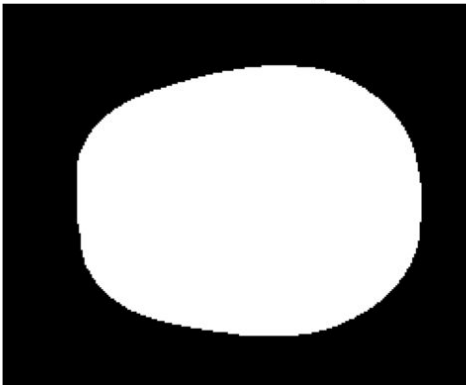
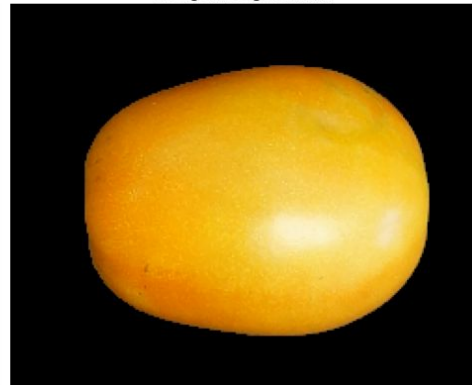
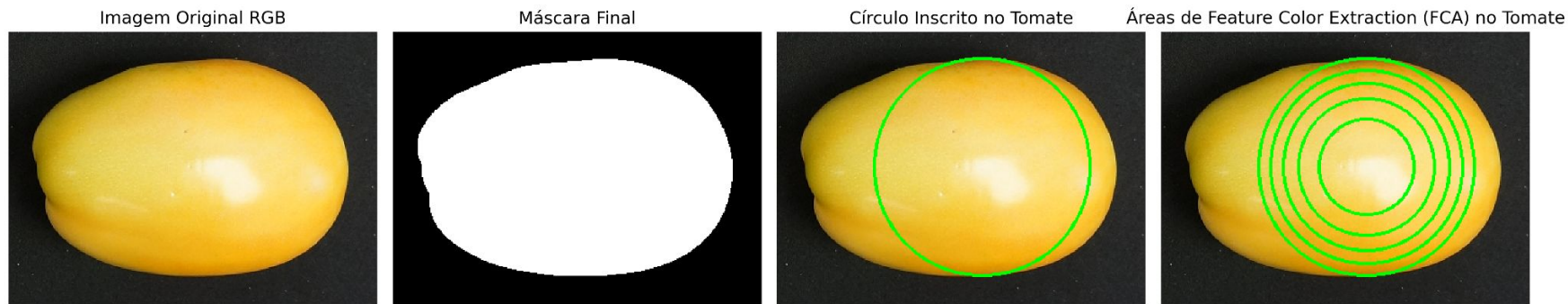
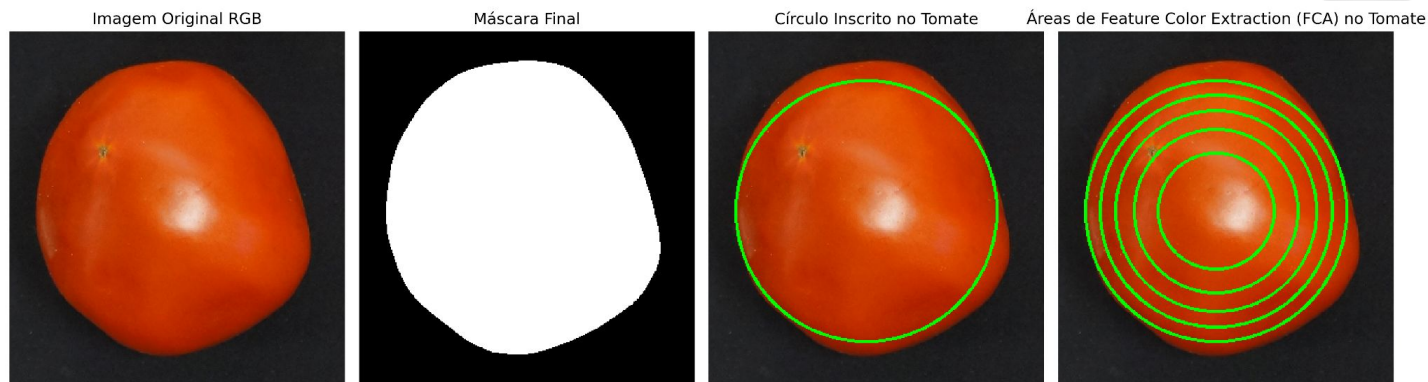


Imagem Segmentada



Segmentação de tomates - Resultado final com Thresholding Adaptativo

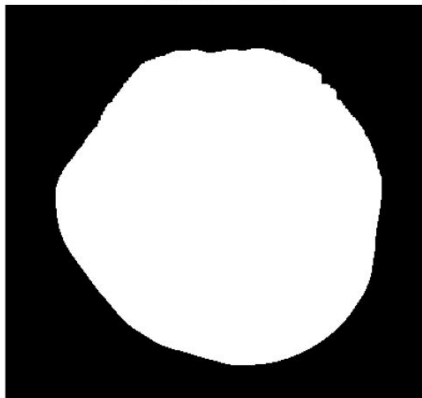


Segmentação de tomates - Resultado final com Thresholding Adaptativo

Imagem Original RGB



Máscara Final



Círculo Inscrito no Tomate



Áreas de Feature Color Extraction (FCA) no Tomate

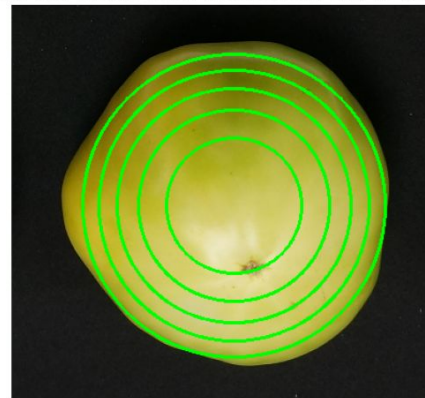
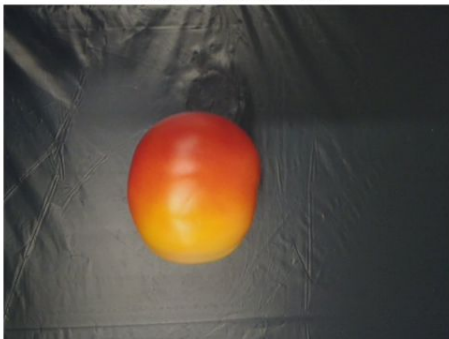
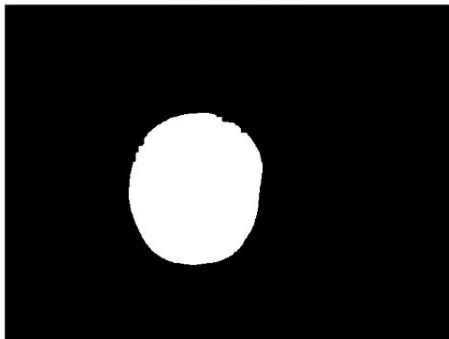


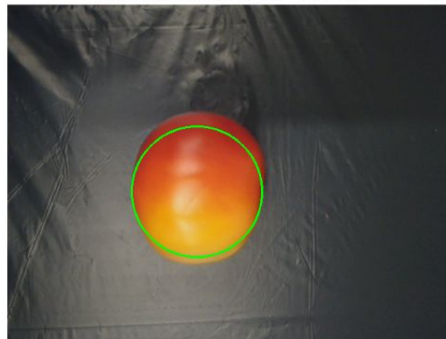
Imagem Original RGB



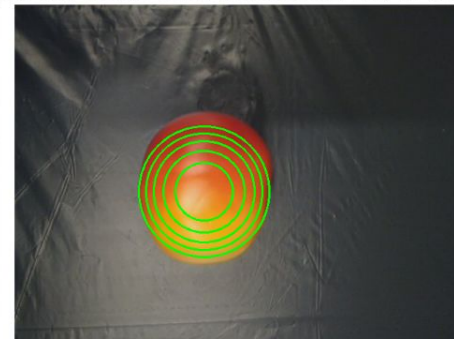
Máscara Final



Círculo Inscrito no Tomate



Áreas de Feature Color Extraction (FCA) no Tomate



Segmentação de tomates - Análise dos Labels do Dataset



(G) Green



(C) Carrot

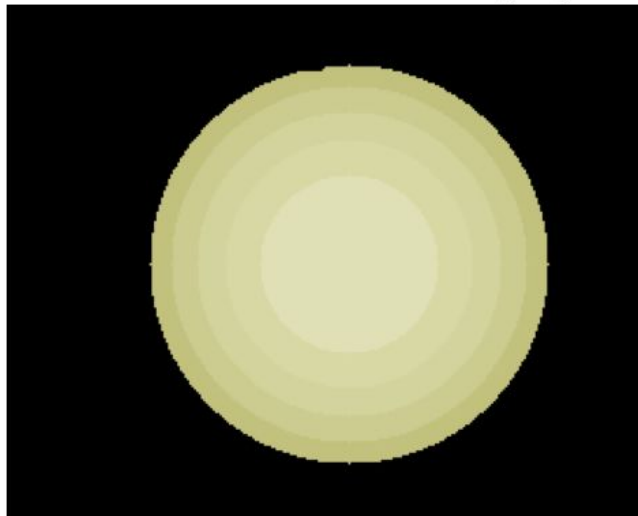


(R) Red

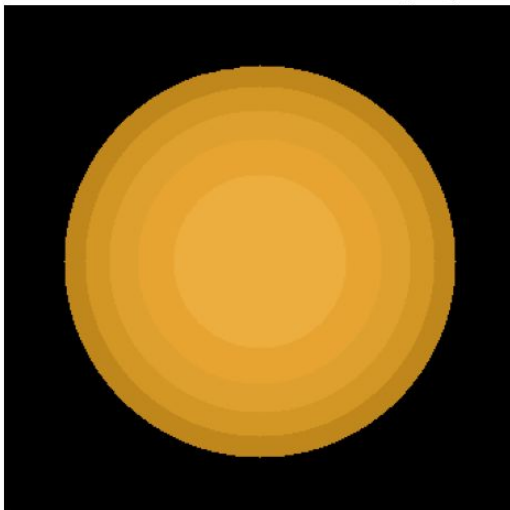


Segmentação de tomates - Entrada da Rede Neural

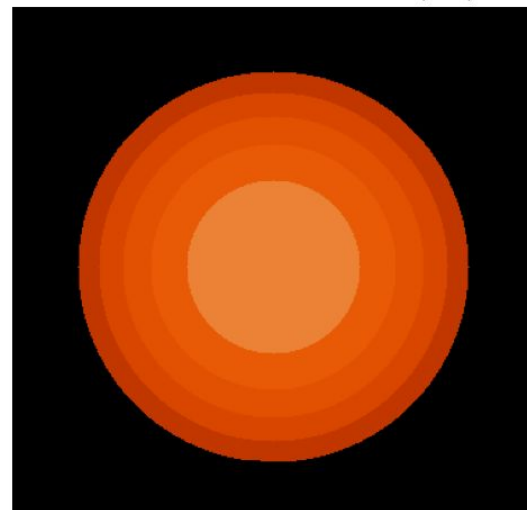
Anéis Pintados com as Cores Médias (RGB)



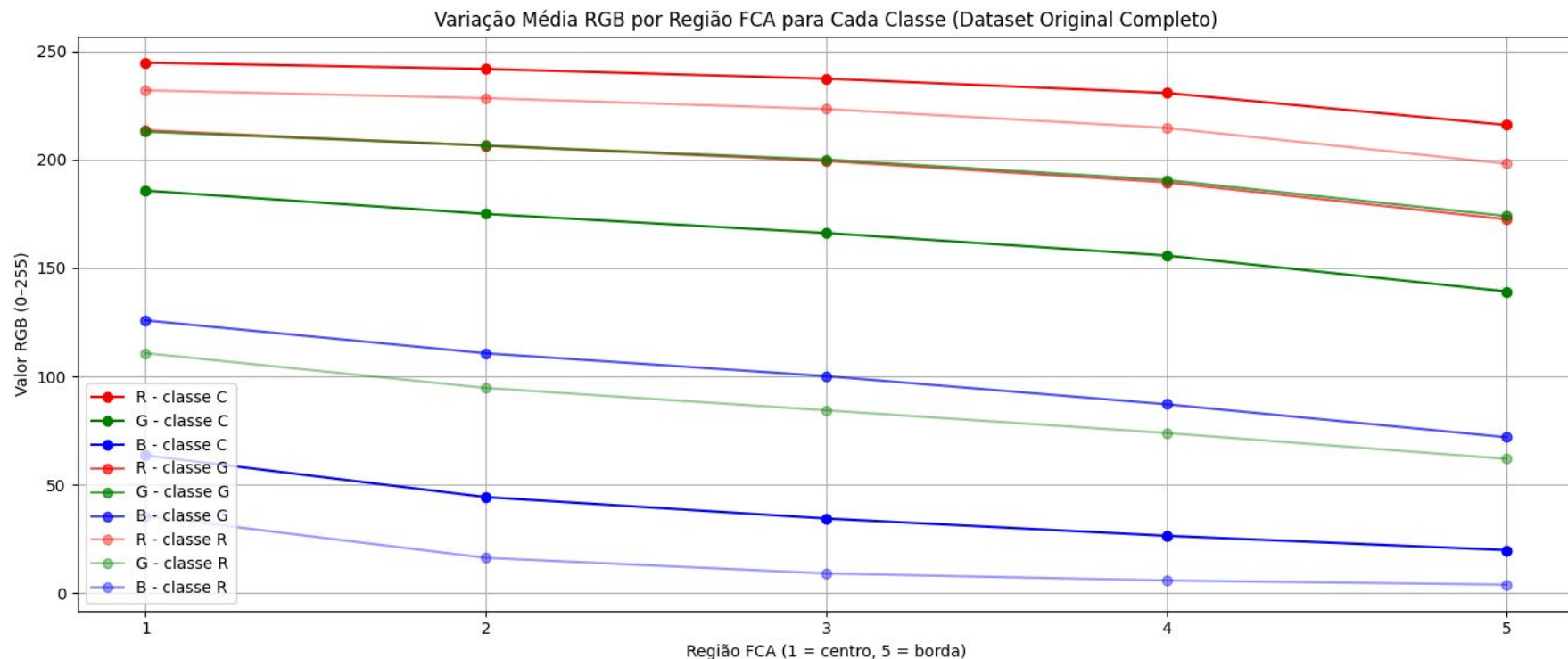
Anéis Pintados com as Cores Médias (RGB)



Anéis Pintados com as Cores Médias (RGB)



Segmentação de tomates - Interpretação dos Resultados

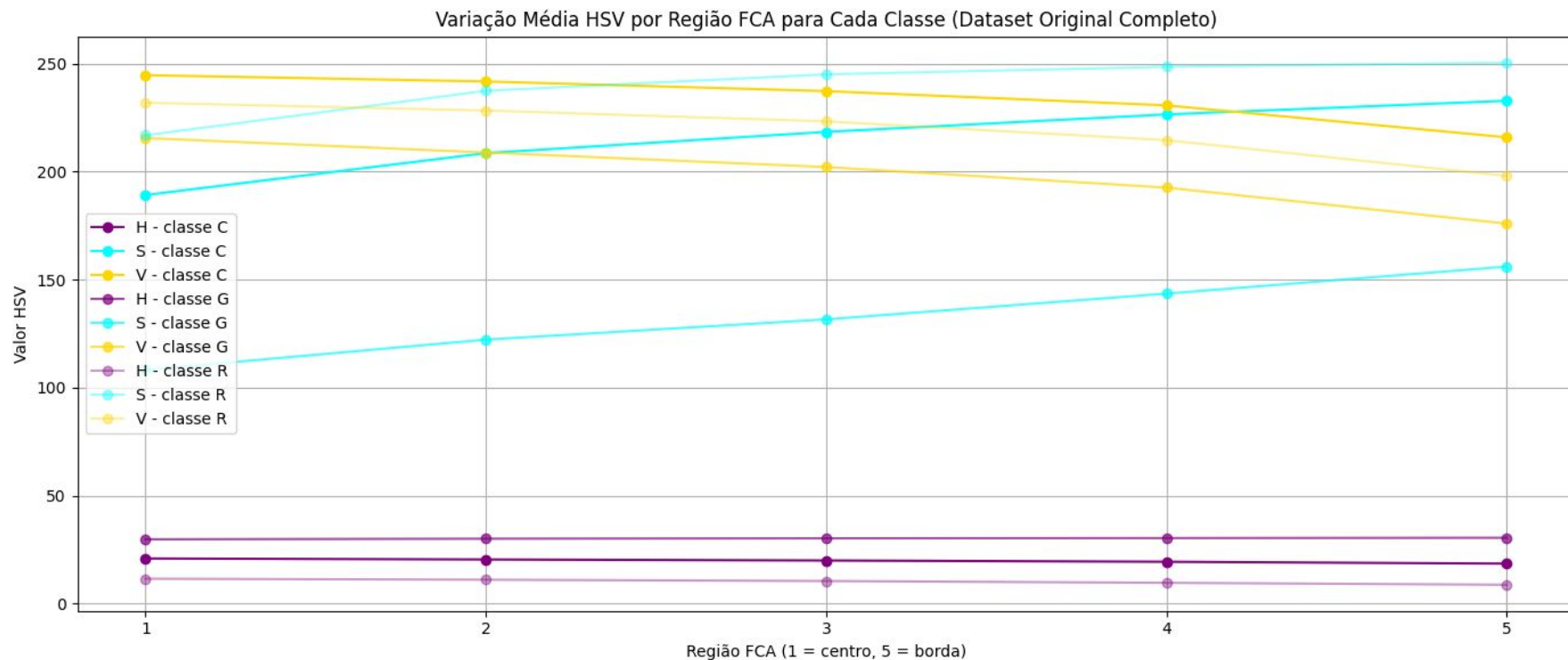


Valores RGB com amostras do dataset original



UNIVERSIDADE
FEDERAL DO CEARÁ

Segmentação de tomates - Interpretação dos Resultados

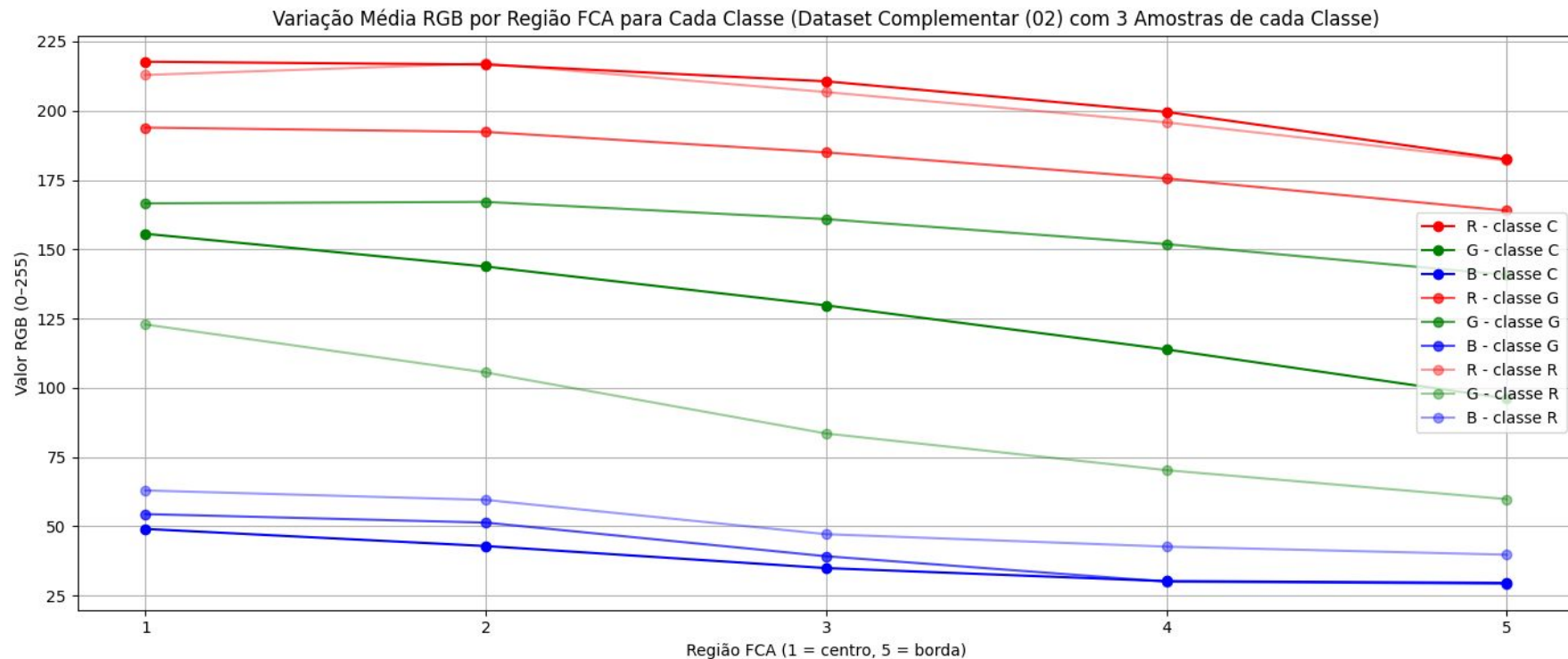


Valores HSV com amostras do dataset original



UNIVERSIDADE
FEDERAL DO CEARÁ

Segmentação de tomates - Interpretação dos Resultados

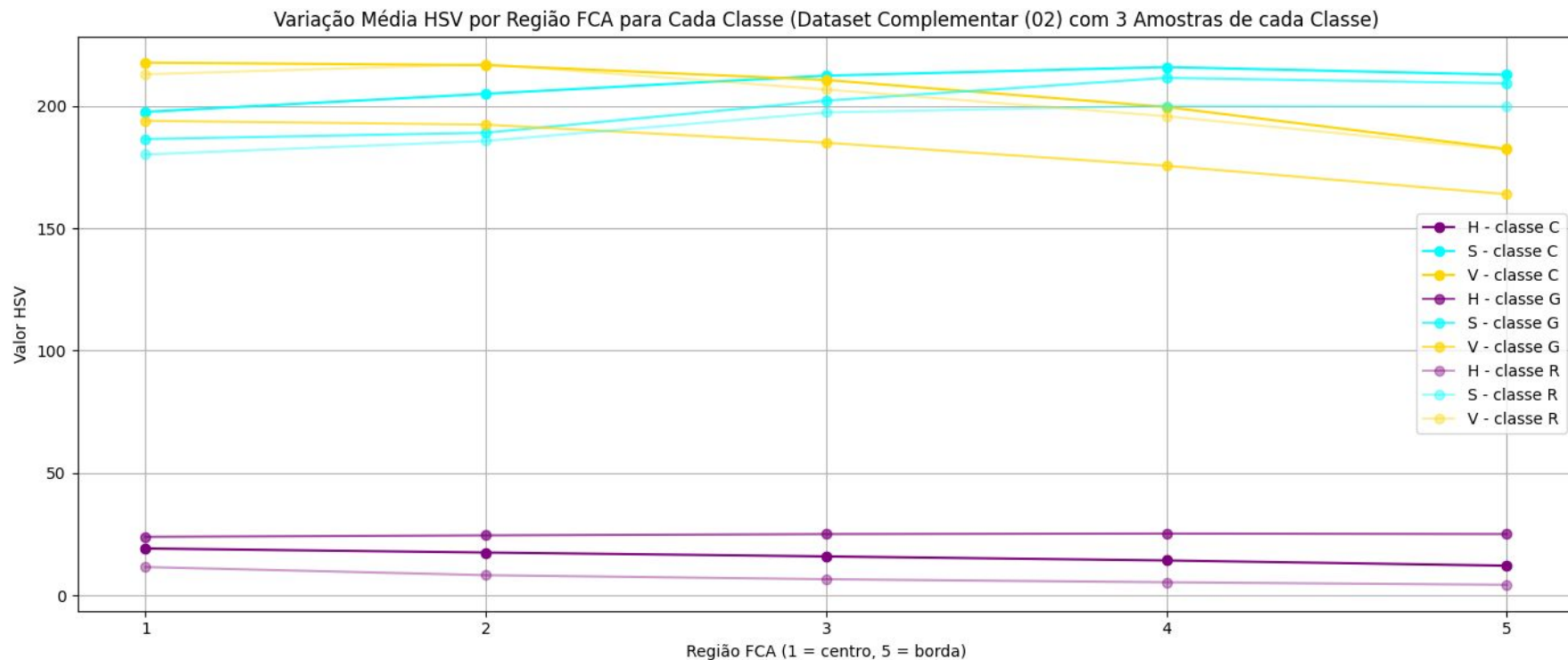


Valores RGB com amostras do dataset 02



UNIVERSIDADE
FEDERAL DO CEARÁ

Segmentação de tomates - Interpretação dos Resultados

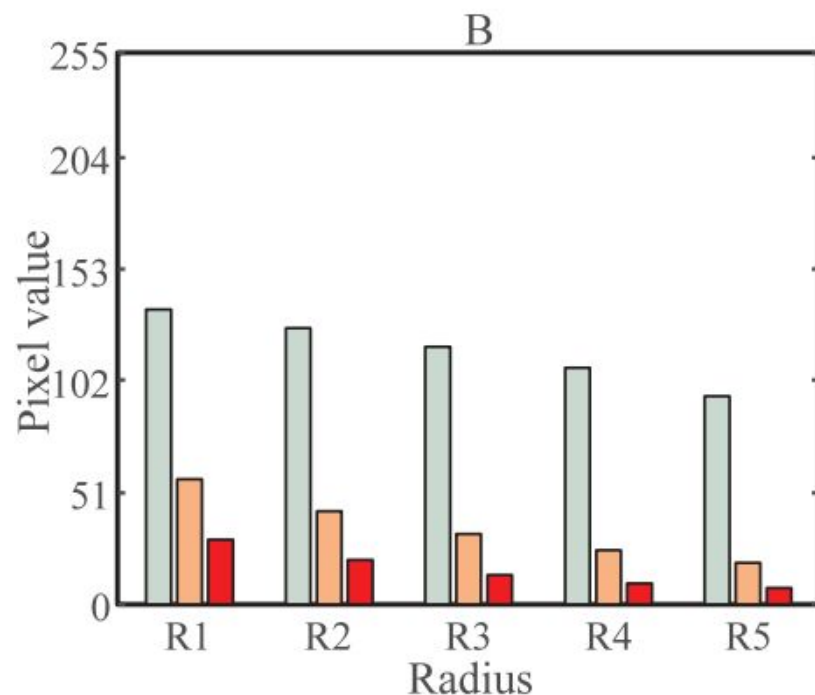
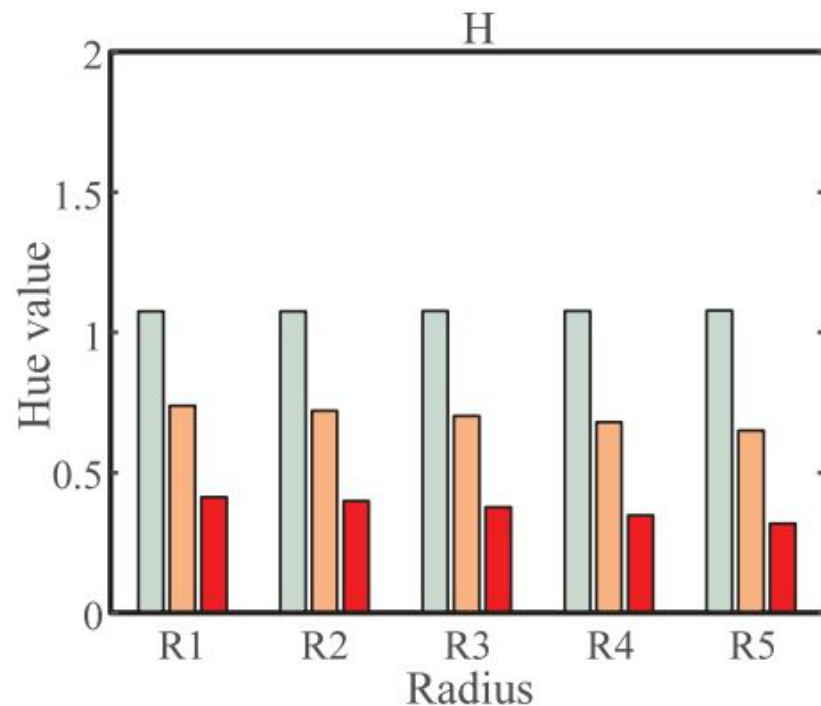


Valores HSV com Amostras do dataset 02

MODELO MLP DE CLASSIFICAÇÃO

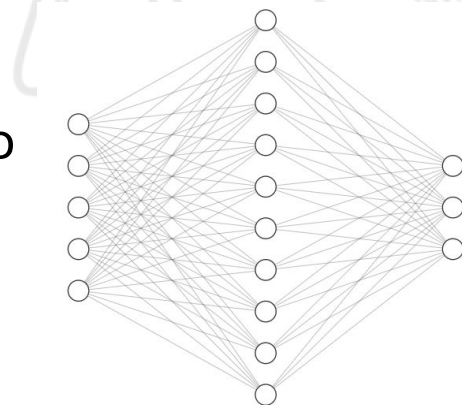
- Modelos Multi-Layer Perceptron (MLP) com Backpropagation são redes neurais artificiais capazes de resolver problemas de classificação e regressão complexos através do ajuste de seus pesos com backpropagation.
- Mais eficientes e flexíveis se comparadas a outras técnicas da época como Análise Discriminante Tradicional e Regressões Logísticas Múltiplas
- A utilização de MLPs para esse problema de **classificação** permite identificar o estágio de maturação dos tomates aprendendo os padrões complexos com base nas características de cor extraídas.
- **Bibliotecas Relacionadas:** Scikit-Learn, Pytorch e Numpy

MLP DE CLASSIFICAÇÃO - Arquitetura do modelo MLP

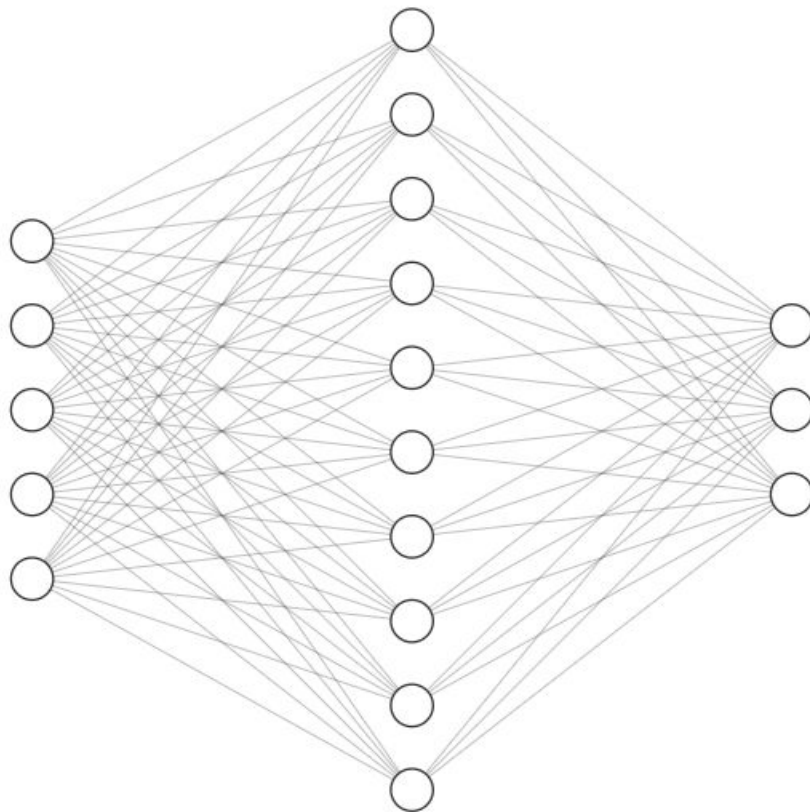


MLP DE CLASSIFICAÇÃO - Arquitetura do modelo MLP

- **Entrada:** 5 neurônios (Valores de **Hue** ou **Blue** das **5 FCAs**)
- **Camada oculta:** 10 neurônios com ativação ReLU
- **Saída:** 3 neurônios com ativação Softmax
 - Saídas: **R** = vermelho, **C** =laranja/Meio-termo



MLP DE CLASSIFICAÇÃO - Arquitetura do modelo MLP



MLP DE CLASSIFICAÇÃO - Código

```
file_name,type,class,R1,G1,B1,R2,G2,B2,R3,G3,B3,R4,G4,B4,R5  
B-C-1.bmp,B,C,236.03,174.73,64.2,229.53,164.11,49.39,220.02  
B-C-2.bmp,B,C,244.29,174.19,66.7,241.24,166.53,48.36,235.07  
B-C-3.bmp,B,C,230.99,203.65,77.49,230.57,194.38,57.07,229.0  
B-G-1.bmp,B,G,183.44,183.47,89.01,173.74,172.0,72.13,167.85  
B-G-10.bmp,B,G,214.2,215.41,117.9,202.45,205.75,88.52,194.1
```



```
X = df[['H1', 'H2', 'H3', 'H4', 'H5']].values  
y = df['class'].values
```



MLP DE CLASSIFICAÇÃO - Código

Codifica a classe de 'R' 'G' para '1','2' etc

```
▶ y = df['class'].values  
  
# Converte rótulos de string para números  
encoder = LabelEncoder()  
y_encoded = encoder.fit_transform(y)
```

```
[93] scaler = StandardScaler()  
X_scaled = scaler.fit_transform(X)  
### Normalização, acho que nem precisava porque os dados são todos da mesma fe
```



MLP DE CLASSIFICAÇÃO - Código

```
▶ X_train, X_test, y_train, y_test = train_test_split(  
    X_scaled, y_encoded, test_size=0.7, random_state=42  
)
```

```
▶ model = MLPClassifier(  
    hidden_layer_sizes=(10), # ← Camadas ocultas (Sabemos que 10 é o total na re  
    activation='relu',      # ← Função de ativação # talvez alterar para sig  
    solver='adam',          # ← Algoritmo de otimização  
    max_iter=500,           # ← Número máximo de épocas (iterações)  
    random_state=42         # ← Semente para reprodução de resultados  
)
```



RESULTADOS

Hue and Blue:
Precisão 56%

	precision	recall	f1-score	support
C	0.33	0.33	0.33	3
G	1.0	0.33	0.5	3
R	0.6	1.0	0.75	3
macro avg	0.64	0.56	0.53	9
weighted avg	0.64	0.56	0.53	9



RESULTADOS

Relatório de Classificação:
Precisão 94%

	precision	recall	f1-score	support
C	1.0	0.71	0.83	17
G	0.97	1.0	0.99	38
R	0.89	1.0	0.94	34
macro avg	0.96	0.9	0.92	89
weighted avg	0.95	0.94	0.94	89



RESULTADOS

Relatório de Validação (Outro dataset:)
Precisão 78%

	precision	recall	f1-score	support
C	1.0	0.71	0.83	17
G	0.97	1.0	0.99	38
R	0.89	1.0	0.94	34
macro avg	0.96	0.9	0.92	89
weighted avg	0.95	0.94	0.94	89

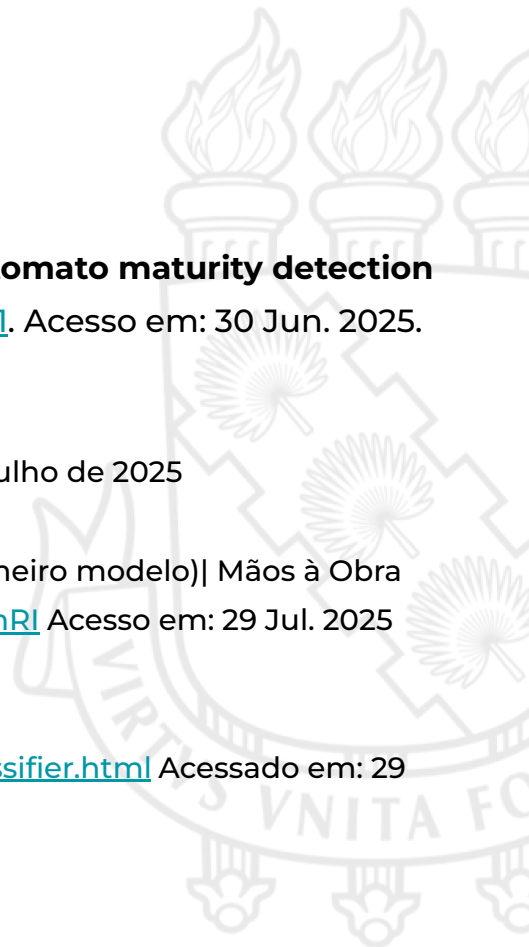


REFERÊNCIAS

Artigo Principal:

Peng Wan, Arash Toudeshki, Hequn Tan, Reza Ehsani. **A methodology for fresh tomato maturity detection using computer vision**. Disponível em: <https://doi.org/10.1016/j.compag.2018.01.011>. Acesso em: 30 Jun. 2025.

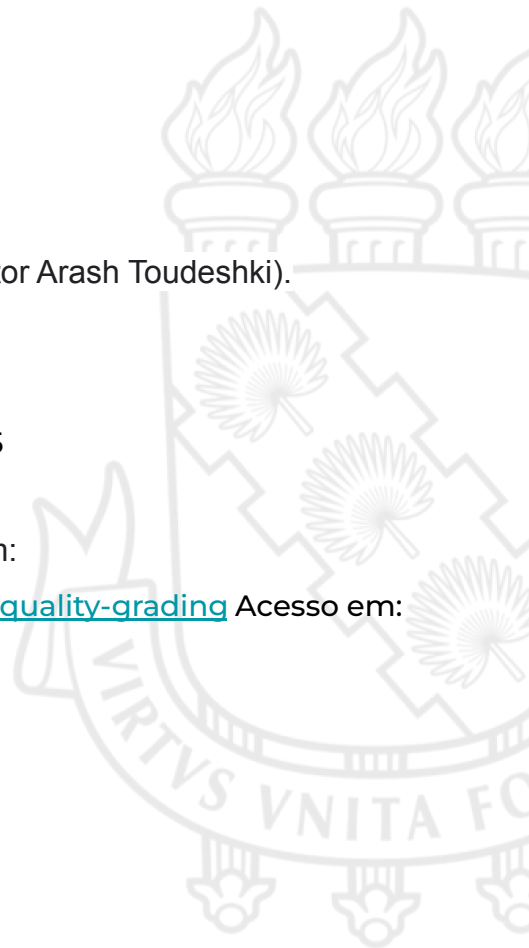
- IBM, O que é segmentação de imagem? Disponível em: <https://www.ibm.com/br-pt/think/topics/image-segmentation> Acesso em: 22 de Julho de 2025
- Programação Dinâmica, Introdução ao Aprendizado de Máquina(do ZERO ao primeiro modelo)| Mãos à Obra Cientista de Dados #01. Disponível em: <https://www.youtube.com/live/QfYGV064mRI> Acesso em: 29 Jul. 2025
- Scikit Learn, documentação MLPClassifier, Disponível em: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html Acessado em: 29 Jul. 2025



REFERÊNCIAS

Datasets Utilizados:

- 1) Dataset fornecido mediante pedido ao autor do artigo principal (autorizado pelo autor Arash Toudeshki).
- 2) Karthik Vinayan, **Tomato fruits dataset**. Disponível em:
<https://www.kaggle.com/datasets/nexuswho/tomatofruits>. Acesso em: 03 Jul. 2025
- 3) Sujay Kapadnis, Tomato Maturity Detection and Quality Grading 📷. Disponível em:
<https://www.kaggle.com/datasets/sujaykapadnis/tomato-maturity-detection-and-quality-grading> Acesso em:
03 Jul. 2025



OBRIGADO PELA ATENÇÃO!

Acesse o nosso trabalho através do link
abaixo ou do QR code ao lado

[https://github.com/CizeLucas/Trabalho
-SeminariosDeComputacao](https://github.com/CizeLucas/Trabalho-SeminariosDeComputacao)

