

## Relatório do Classificador de Frutas – Processamento Digital de Imagens

Alunos: Fernando Luiz, Iuri Schmoeller e Pedro Horst  
Professor: Fábio Spanhol  
Data: 01/12/2023

## Sumário

1 Introdução.....	1
2 Projeto no GitHub.....	1
3 Criação da base de dados.....	1
4 Data annotation.....	2
5 Data augmentation.....	2
6 Data normalization.....	2
7 Segmentação dos objetos.....	3
8 Classificador.....	3
9 Resultados.....	3
Links.....	4

## 1 Introdução

O presente relatório tem como objetivo explicar e contextualizar a implementação de um classificador de frutas, enfatizando os processos utilizados durante o trabalho.

## 2 Projeto no GitHub

O projeto completo encontra-se no diretório do GitHub [1] e está dividido em duas pastas, “Images” que contem as imagens reais e as geradas para expandir a base de dados, “Masks”, relativa as máscaras obtidas (ground truth), além de um arquivo, “Masks\_Generator.py”, que contempla a implementação de um algoritmo para obter as máscaras. Um notebook, “FruitClassifier.ipynb”, que apresenta os scripts utilizados na manipulação das imagens e a implementação do classificador, além de um arquivo de metadados “README.md”, que apresenta uma descrição do projeto e os IDs das frutas utilizadas.

## 3 Criação da base de dados

A base de dados utilizada possui 10 classes de frutas, descritas na tabela 1, sendo utilizado as imagens originais e as imagens modificadas através de filtros.

ID	Classe da fruta
0	Banana
1	Clementine

2	Lemon
3	Tomato
4	Strawberry
5	Apple
6	Pineapple
7	Pear
8	Papaya
9	Coconut

---

Tabela 1 – Frutas e seus respectivos Ids

#### 4 Data annotation

Para o bounding box utilizamos o aplicativo web CVAT [2], porém acabamos não utilizando no algoritmo de classificação, e por isso não incluímos no trabalho.

#### 5 Data augmentation

Para a expansão da base de dados utilizamos o script que encontra-se no endereço, `FruitClassifier/Images/DataAugmentation.py`, ele engloba as 3 funções principais, logaritmo da imagem, exponencial da imagem e o filtro da média. As imagens podem ser vistas na Figura 1.

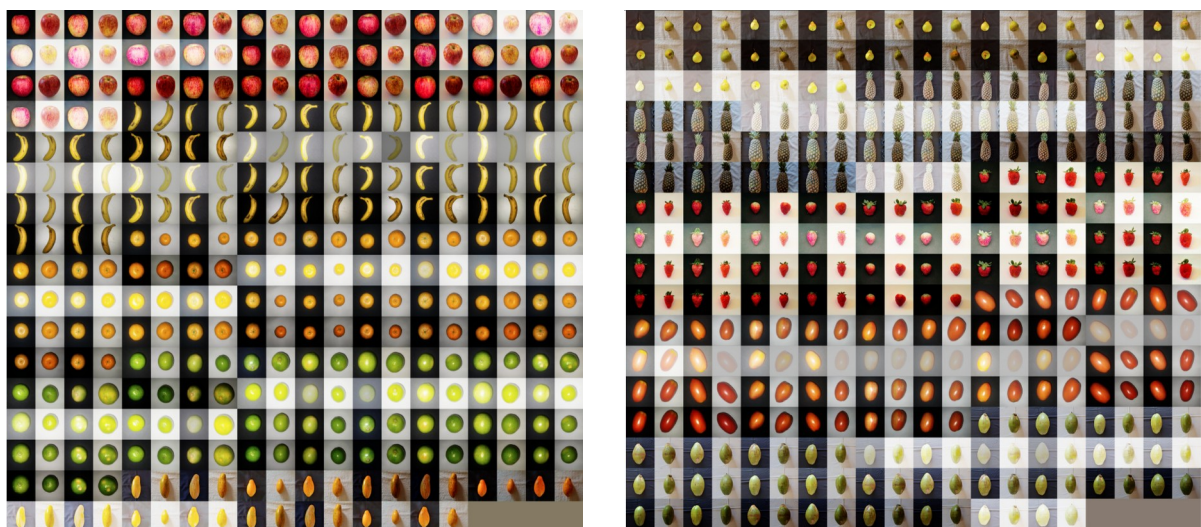


Figura 1 – Imagens originais e geradas no data augmentation

#### 6 Data normalization

Para os dados normalizados foi utilizado o algoritmo de equalização dos histogramas das imagens, e em cima das imagens normalizadas, foram aplicados a média e os respectivos histogramas de cada canal RGB das imagens médias, isso pode ser visto na Figura 2.

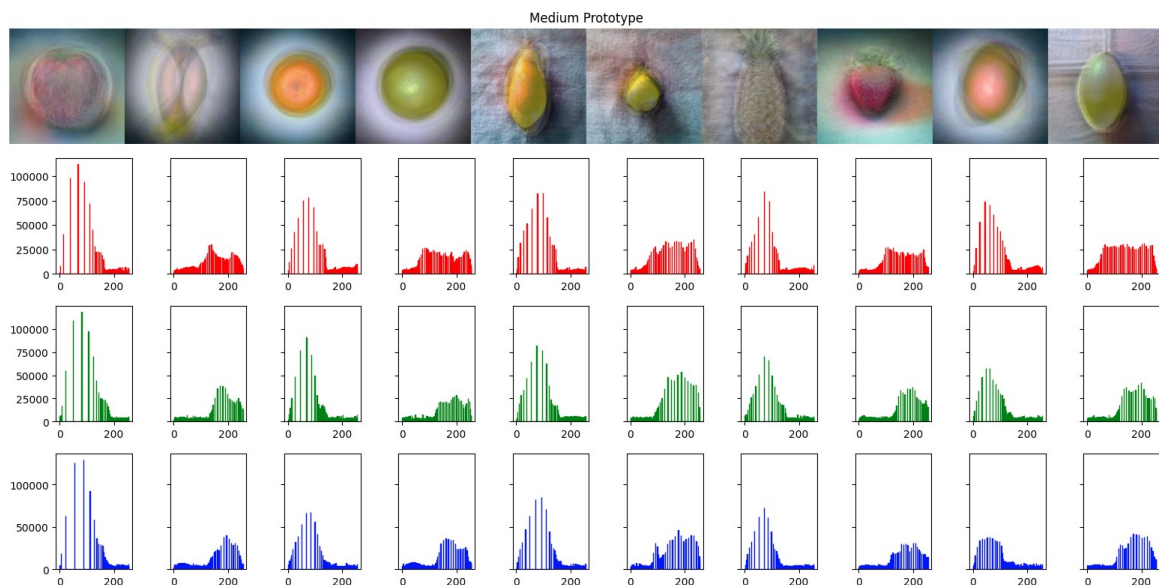


Figura 2 – Imagens médias e os respectivos histogramas

## 7 Segmentação dos objetos

Para a obtenção do ground truth foi utilizado a abordagem manual para imagens que apresentavam muitos ruídos, e abordagem automática através do código presente em `FruitClassifier/Masks/Masks_Generator.py`. As máscaras geradas se encontram na pasta “Masks” do diretório.

## 8 Classificador

Para a tarefa de classificação das imagens, optamos por utilizar a arquitetura de rede neural convolucional (CNN) ResNet-50. Esta escolha se baseia na capacidade comprovada das CNNs em lidar eficazmente com dados de imagens, explorando filtros convolucionais para extrair características importantes.

As imagens foram divididas em 70% para treinamento, 15% para teste e os outros 15% para validar o modelo, além de que a rede é pré-treinada com pesos obtidos a partir da base de dados ImageNet.

## 9 Resultados

As Figura 3 e Figura 4 mostram as métricas obtidas no modelo e a matriz de confusão respectivamente. Observa-se uma alta precisão do modelo chegando a 99%, porém como a base de dados é consideravelmente pequena seria necessário mais imagens para verificar a verdadeira acurácia do modelo. Para a base de dados criados os resultados foram satisfatórios, ocorrendo somente um erro, onde ele classificou uma *clementine* como *lemon*.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	9
1	1.00	0.89	0.94	9
2	0.94	1.00	0.97	16
3	1.00	1.00	1.00	18
4	1.00	1.00	1.00	6
5	1.00	1.00	1.00	10
6	1.00	1.00	1.00	9
7	1.00	1.00	1.00	7
8	1.00	1.00	1.00	8
9	1.00	1.00	1.00	9
accuracy			0.99	101
macro avg	0.99	0.99	0.99	101
weighted avg	0.99	0.99	0.99	101

Figura 3 – Avaliação do modelo

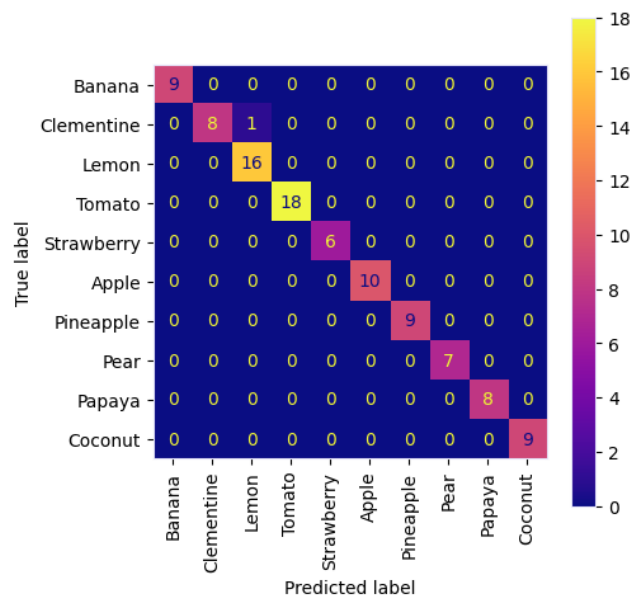


Figura 4 – Matriz de confusão do modelo

## Links

- [1] <https://github.com/schmoellerluri/FruitClassifier.git>
- [2] <https://app.cvat.ai/>