

# Classificação de Emoções do Rosto humano

---



Equipe 5 - Balduino da Silva | Cassia Santos | Francisco Expedito | Matheus  
Soares | Thayane Gonçalo | Giulia Duarte | Lucas Almeida

# Sumário

---

- Introdução
- Materias e Métodos
- Resultados
- Conclusões

# Introdução

---

A base de dados "Human Face Emotions" disponível no Kaggle é uma coleção de imagens de rostos humanos classificadas em diferentes expressões faciais.

A base é amplamente utilizado em projetos de Reconhecimento de Emoções com CNNs, Aplicações em Tempo Real e Modelos de Transfer Learning.



# Objetivo

---

O objetivo geral consiste em treinar um modelo que identifica e classifica as emoções de alegria, tristeza e raiva dos humanos a partir de imagens.

O objetivo específico consiste em treinar o modelo EfficientNet proposto pela referência, de forma a melhorar a métrica de acurácia do modelo.

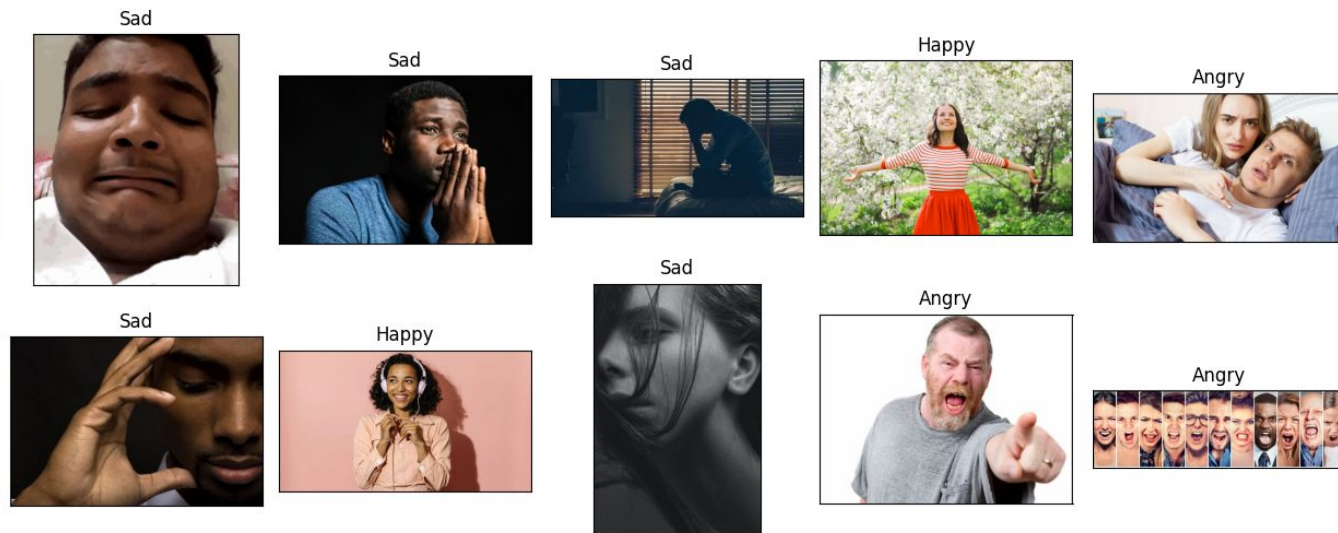


# **Materiais e Métodos**

---

# Base de Dados

Este conjunto de dados contém imagens de rostos humanos com emoções felizes, tristes e raivosos.

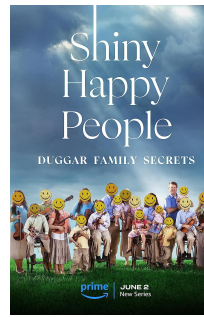


# Base de dados

A base de dados possui algumas imagens corrompidas, duplicadas e inadequada para o modelo.

Tabela 1: Qualidade da imagem

| Classes | Corrompidas | Duplicados | Inadequadas |
|---------|-------------|------------|-------------|
| Angry   | 6           | 4          | 2           |
| Happy   | 1           | 6          | 8           |
| Sad     | 5           | 0          | 10          |
| Total   | 12          | 10         | 20          |



# Base de Dados

---

Classe raiva:



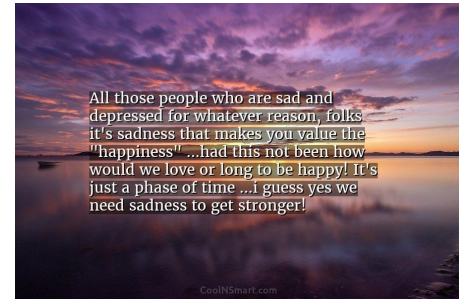
shutterstock.com - 292800785



shutterstock.com - 292800785



Classe tristeza:





# Modelo Efficientnetb0

---

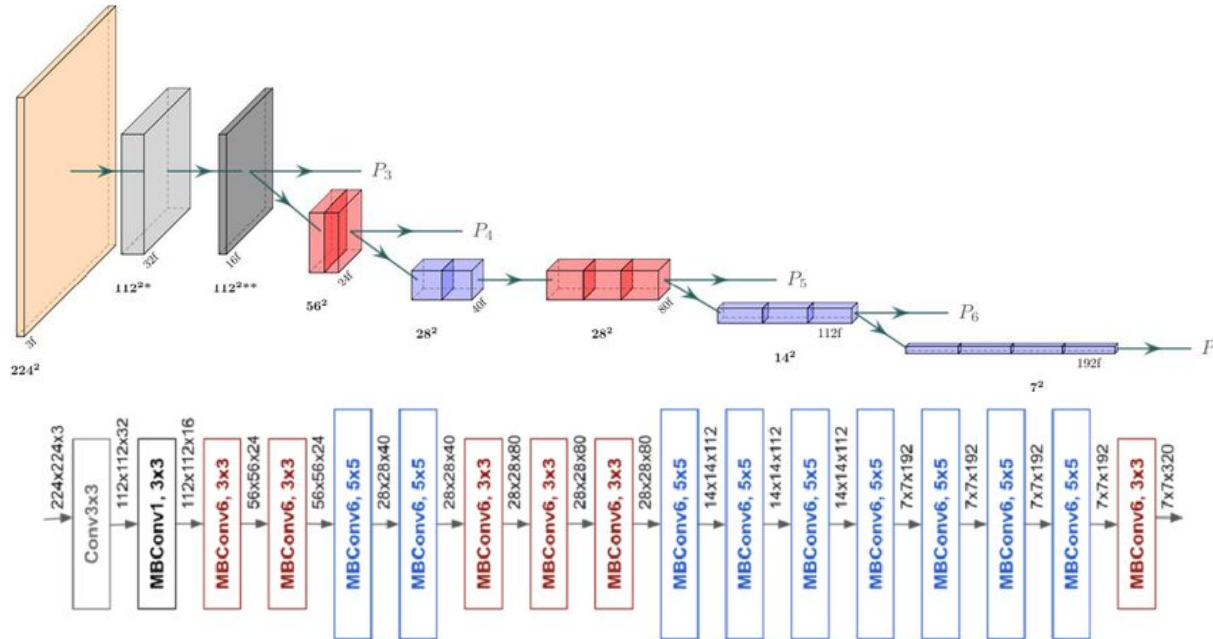
Disponível no Kaggle nomeado de “human-emotions-efficientnetb0” do autor Jovan Chua e possui a acurácia de 67%.

Vantagens:

- Configura um pipeline completo para treinamento e avaliação de um modelo de deep learning usando PyTorch.
- Define um dataset personalizado
- Define funções para treinar e avaliar o modelo
- Utiliza métricas para acompanhar o desempenho durante o processo.

# Modelo

O modelo utilizado é a EfficientNet\_B0 da biblioteca torchvision models, que possui como arquitetura base a ConvNet, sendo otimizada para eficiência e desempenho.



# Modelo

---

1. Parâmetros do Modelo:
  - Resolução da Imagem: 224x224 pixels.
  - Número de Camadas: 82.
  - Aproximadamente 5.3 milhões.
  - Parâmetros Treináveis: Sim, mas ajustável para congelar camadas.
2. Principais Características:
  - Módulos MBConv: 7 blocos com convoluções expansivas.
  - Ativação: SiLU (Swish).
  - Pooling: Adaptive Average Pooling.
3. Camada Final: Fully connected para classificação.

# Modelo

O modelo EfficientNet utiliza-se da técnica de “Transfer Learning” , onde os pesos do aprendizado que já foi treinado em um grande conjunto de dados por organizações para uma tarefa específica é carregado no modelo.

| Layer (Type)          | Input Shape        | Output Shape       | Param # | Trainable |
|-----------------------|--------------------|--------------------|---------|-----------|
| Conv2d (features.0.0) | [32, 3, 224, 224]  | [32, 32, 112, 112] | 864     | False     |
| MBConv (features.1.0) | [32, 32, 112, 112] | [32, 16, 112, 112] | 1,448   | False     |
| MBConv (features.4.0) | [32, 40, 28, 28]   | [32, 80, 14, 14]   | 37,130  | False     |
| MBConv (features.6.0) | [32, 112, 14, 14]  | [32, 192, 7, 7]    | 262,492 | False     |
| Conv2d (features.8.0) | [32, 320, 7, 7]    | [32, 1280, 7, 7]   | 409,600 | True      |
| Linear (classifier.1) | [32, 1280]         | [32, 3]            | 3,843   | True      |

# Melhoria do Conjunto de Dados

---

## 1. Limpeza dos Dados:

Remoção de imagens duplicadas e de baixa qualidade.  
Impacto: Reduz ruído, melhora a acurácia do modelo.

## 2. Ampliar o Conjunto de Dados:

Descrição: Adição de imagens em variados cenários.  
Impacto: Enriquecimento do dataset, melhora na distinção de emoções.

# Melhoria do Conjunto de Dados

---

Para ampliar o conjunto de dados, a técnica usada foi a adição de imagens de outra base de dados.

É usado a base de dados do Kaggle FER-2013, que contém imagens de rostos em escala de cinza de 48x48 pixels.



# Otimização do Modelo

---

## 1. Balanceamento das Classes:

- Descrição: Igualar a quantidade de imagens entre classes.
- Impacto: Previne viés, melhora a acurácia em classes menos representadas.

## 2. Redução na Dimensão das Imagens:

- Descrição: Ajuste do tamanho das imagens.
- Impacto: Acelera o treinamento, mantém a eficiência computacional.

# Resultados

---



# Base de dados

---

Após a limpeza, pré-processamento e adição de novas imagens na base de dados. Seu tamanho 720 amostras, com resolução de 48x48 pixels e em escala de cinza.



## Modelo da referência

---

É possível observar que por mais que o modelo aprenda com os dados de treinamento, não é possível generalizar com os dados de teste.

| Época | Perda de Treinamento | Acurácia de Treinamento | Perda de Teste | Acurácia de Teste |
|-------|----------------------|-------------------------|----------------|-------------------|
| 0     | 1.02943              | 0,51%                   | 0.90412        | 0,56%             |
| 1     | 0.73483              | 0,74%                   | 0.76880        | 0,74%             |
| 2     | 0.54129              | 0,81%                   | 0.67089        | 0,73%             |
| 3     | 0.43293              | 0,90%                   | 0.63564        | 0,70%             |
| 4     | 0.35312              | 0,92%                   | 0.62396        | 0,72%             |
| 5     | 0.29912              | 0,94%                   | 0.66314        | 0,68%             |

# Modelo Proposto

---

Já no modelo proposto é possível observar uma melhora na sua generalização.

| Época | Perda de Treinamento | Acurácia de Treinamento | Perda de Teste | Acurácia de Teste |
|-------|----------------------|-------------------------|----------------|-------------------|
| 0     | 0.16324              | 0,96%                   | 0.36805        | 0,85%             |
| 1     | 0.11552              | 0,98%                   | 0.35697        | 0,88%             |
| 2     | 0.11365              | 0,98%                   | 0.34919        | 0,89%             |
| 3     | 0.09538              | 0,98%                   | 0.34884        | 0,89%             |
| 14    | 0.02551              | 0,99%                   | 0.35924        | 0,88%             |
| 15    | 0.03168              | 0,99%                   | 0.36878        | 0,89%             |
| 16    | 0.02706              | 0,99%                   | 0.38192        | 0,89%             |
| 17    | 0.03210              | 0,99%                   | 0.40462        | 0,89%             |
| 18    | 0.03490              | 0,99%                   | 0.41144        | 0,89%             |
| 19    | 0.03006              | 0,99%                   | 0.40458        | 0,89%             |

# Modelos

Essa comparação destaca que o Modelo Proposto tem uma acurácia média significativamente melhor em comparação com o Modelo Referência.

| Modelo            | Média da Acurácia de Treinamento | Média da Acurácia de Teste |
|-------------------|----------------------------------|----------------------------|
| Modelo Referência | 0,79%                            | 0,69%                      |
| Modelo Proposto   | 0,98%                            | 0,88%                      |

## Conclusões

---

- A base de dados possui amostras duplicadas e inadequadas para o uso no modelo.
- Por mais que a base de dados se proponha a classificar expressões faciais, na prática as imagens comportam também expressões corporais. Levando a concluir que elas dificultava a classificação das emoções faciais.



# Referências

---

<https://www.kaggle.com/code/jovanchua/human-emotions-efficientnetb0>

<https://www.kaggle.com/datasets/msambare/fer2013>

—

**Obrigado!**