

# Como usar a IA

contatos para pegar a IA ou tirar dúvidas:

GitHub: [https://github.com/mathreurs/Scripts\\_AI](https://github.com/mathreurs/Scripts_AI)

Número: (27)996965998

## Ressalvas:

No GitHub existem mais de uma pasta com o nome de IA, nos vamos usar a **AI5**.

A versão do python é a 3.9 e precisa das seguintes bibliotecas:

- Mediapipe
- Opencv
- OS
- Numpy
- Pickle
- Matplotlib
- Sklearn

Toda podem ser instaladas em suas últimas versões.

Por favor não colocar as 2 mãos ao mesmo tempo durante a captura, ela irá parar na mesma hora e fechará o programa.

De preferência usar uma câmera com boa definição, se manter longe de grandes focos de luz e manter apenas 1 pessoa na frente da câmera.

## Nomes dos scripts:

Os scripts descritos abaixo estão em ordem de uso, caso queira treinar a IA para aprender sinais próprios seus.

Caso não queira, basta apenas executar o último script com os sinais já treinados.

### 1. Collect\_img.py

Esse script coleta e armazena as imagens capturadas pela câmera, quando executado ele vai abrir a câmera e aguardar você apertar 'Q' para começar a capturar as imagens.

```
number_of_classes = 5  
dataset_size = 100
```

`Number_of_classes` é o número de sinais diferentes que poderão ser capturados, ao fim da captura uma mensagem irá aparecer na tela indicando que já pode iniciar uma nova apertando 'Q'.

`Dataset_size` é o número de imagens que serão capturadas por cada pasta de sinais.

### 2. Create\_dataset

Ao executar esse script apenas o espere terminar a execução que será criado um arquivo com o nome de `DATA.PIKLE`, assim que esse arquivo for criado poderá ir para o próximo passo.

```
create_dataset.py  
data.pickle
```

### 3. Train\_classifier

Executando esse script será criado um arquivo chamado DATA.P, quando ele for criado basta apenas ir para o próximo, e último, script.

```
model.p  
train_classifier.py
```

### 4. Interface\_classifier

Para dar nome para os sinais que você tenha treinado deverá abrir o script e trocar na seguinte variável:

```
labels_dict = {0: 'A', 1: 'B', 2: 'C', 3: 'D', 4: 'E'}
```

Ela segue a ordem de captura da câmera, bastando

Ao executar ele irá abrir uma tela com a imagem da câmera, bastando apenas fazer os sinais um uma das mãos que ele irá interpretar.

Sinais previamente treinados:

A:



B:



C:



D:



E:

