

Detecção de instrumentais cirúrgicos

Índice

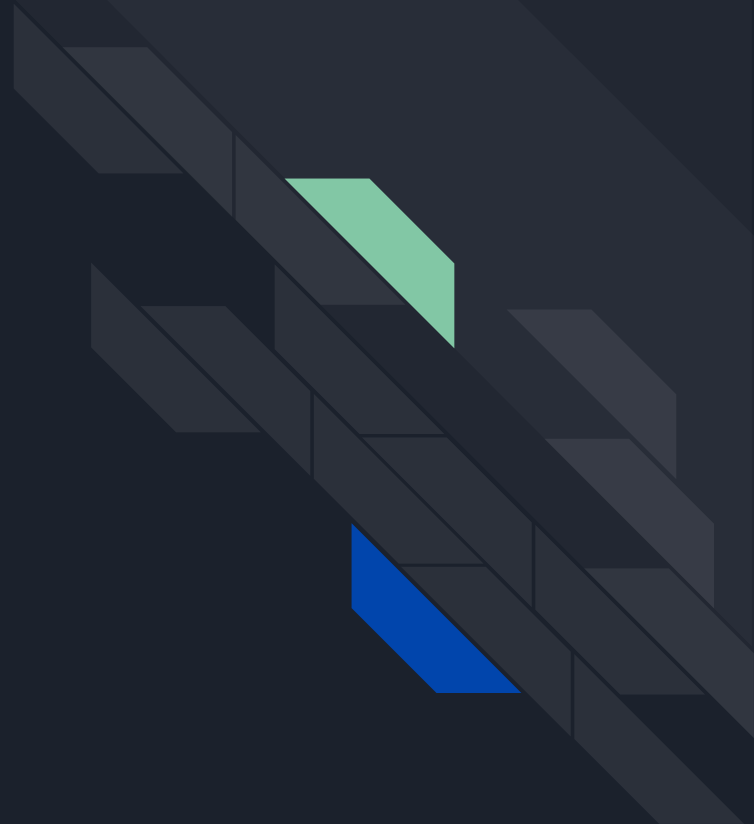
Visão geral

Entender seus problemas

Objetivo do projeto

Procedimentos

Resultado

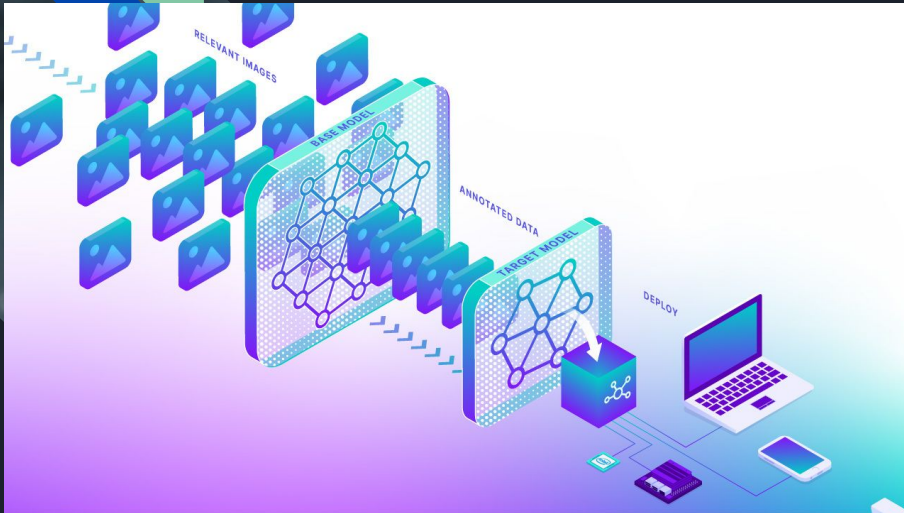




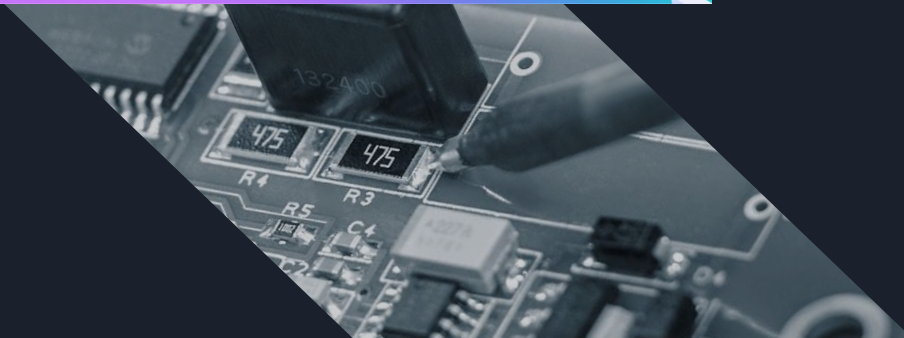
Procedimentos

- 01 Criação do dataset, utilizando um modelo de reconhecimento como base
- 02 Organização do dataset no formato YOLO utilizando algoritmos em Python
- 03 Treinamento do modelo
- 04 Análise dos resultados

Criação do Dataset (Auto Distill)




Para a criação do dataset, foi utilizado de inspiração os conceitos do módulo Auto Distill, que consiste em treinar modelos sem precisar rotular um grande número de imagens. Utilizando modelos base para realizar a rotulação.





Criação do Dataset

Foi realizado um treinamento de um modelo com uma classe única para servir de base para a geração do dataset principal, esse modelo identifica o instrumental e retorna as coordenadas da bounding box. A partir desse modelo base é possível rotular as classes do modelo principal apenas com a detecção do modelo base.






Organização do dataset

Com um algoritmo simples é possível percorrer uma pasta, enviar as imagens para o modelo base e gerar um txt para cada imagem encontrada, com o número da classe e as coordenadas retornadas pelo modelo base.

Com outro algoritmo é possível organizar as imagens de todas as classes juntas separadas por pastas de train, test e valid e gerar o arquivo data.yaml, que conterà o caminho das pastas, número e nome das classes e parâmetros de treinamento

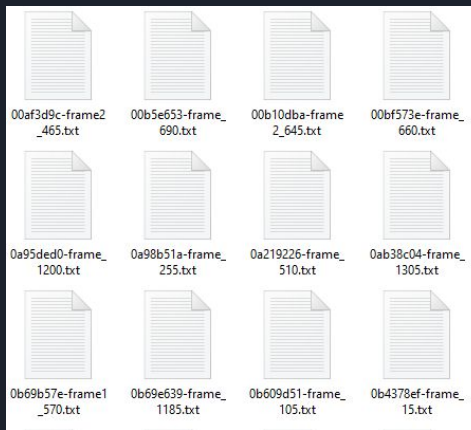


Estrutura do dataset

 train	24/05/2024 13:53	Pasta de arquivos	
 valid	24/05/2024 13:54	Pasta de arquivos	
 data.yaml	24/05/2024 14:19	Arquivo Fonte Yaml	1 KB



 images	24/05/2024 13:53	Pasta de arquivos	
 labels	24/05/2024 13:53	Pasta de arquivos	



00af3d9c-frame2_465.txt - Bloco de Notas

Arquivo Editar Formatar Exibir Ajuda

0.49126847054308187 0.4758203799654577 0.3745922087890999 0.7616580310880829

Arquivo data.yaml

```
train: D:/detectaInstrumentais.v2i.yolov8/Datasetv5/train  
val: D:/detectaInstrumentais.v2i.yolov8/Datasetv5/valid
```

```
nc: 17
```

```
names: [  
  'Afastador-HOHMANN-21cm',  
  'Afastador-SENN-MILLER-AGUDO-16cm',  
  'Afastador-Weitlaner',  
  'Afastador-taylor',  
  'Cabo-de-bisturi',  
  'Cureta-volkman-21cm-8mm',  
  'Pinca-ANATOMICA-DISSECCAO-14cm',  
  'Pinca-adson-12cm-com-serrilha',  
  'Pinca-anatomica-dente-de-rato',  
  'Pinca-backaus-8cm',  
  'Pinca-cheron-24cm',  
  'Pinca-kelly-16cm-curva',  
  'Pinca-mixer-16cm',  
  'Porta-agulha-MAYO-HEGAR-14cm',  
  'Tesoura-IRIS-12cm-curva',  
  'Tesoura-METZEMBAUM-20cm-curva',  
  'Tesoura-cirurgica-17cm']
```

```
#backbone: 'CSPDarknet53'
```

```
#head: 'YOLOv8Head'
```

```
#model:
```

```
# name: 'yolov8'
```

```
#hsv_h : .015
```

```
#hsv_s : .7
```

```
#hsv_v : .4
```

```
#degrees : .4
```



Caminho para as pastas do dataset



Número de Classes



Nome das Classes



Parâmetros

Treinamento

Com o dataset no formato correto, é possível realizar o treinamento

```
from ultralytics import YOLO
import torch
from ultralytics import NAS
#import super_gradients

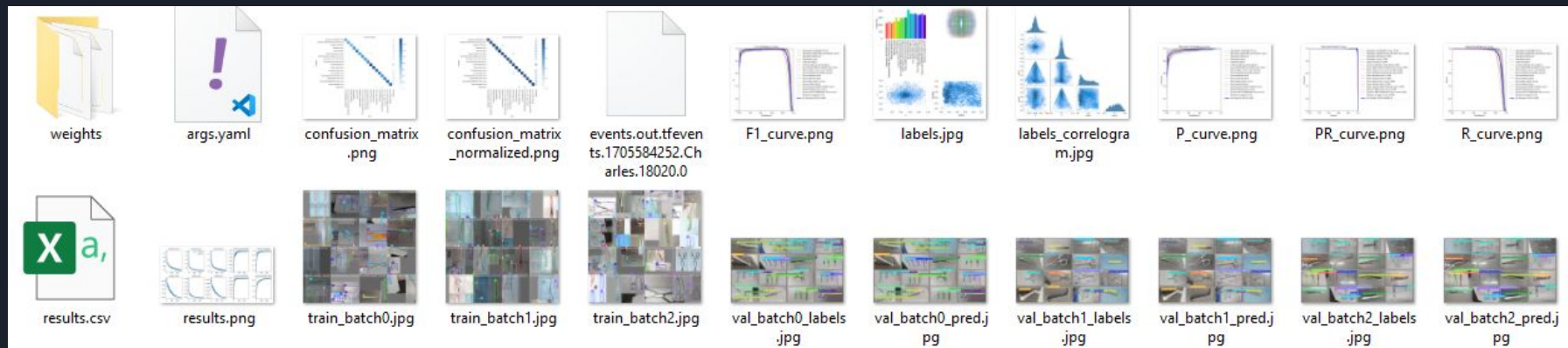
if torch.cuda.is_available():
    torch.cuda.empty_cache()

if __name__ == '__main__':
    #model = YOLO('yolov8n.pt').load('runs/detect/train47/weights/best.pt') # load a pretrained YOLOv8n classification model
    model = YOLO('yolov8n.pt')
    # Load a COCO-pretrained YOLO-NAS-s model
    #model = NAS('yolo_nas_s.pt')
    model.train(
        data='dataset_2802/data.yaml',
        epochs=1000,
        patience=50,
        batch= -1,
        device=0,
        lrf = 0.1,
        #augment=True,
        #hsv_h = .015,
        #hsv_s = .7,
        #hsv_v = .4,
        #degrees = .4,
        #translate = .3,
        #scale = .5,
        #shear = .01,
        #flipud = .3,
        #fliplr = .5,
        #mixup = .5
    )
```


Importamos o modelo padrão do yolo V8 e definimos o dataset, número de épocas e outros parâmetros.

Resultados

Com o treinamento do modelo principal finalizado é criada uma pasta com os pesos gerados e alguns gráficos dos resultados obtidos no treinamento.




Pesos retornados pelo modelo:

 **best.pt**

18/01/2024 19:07

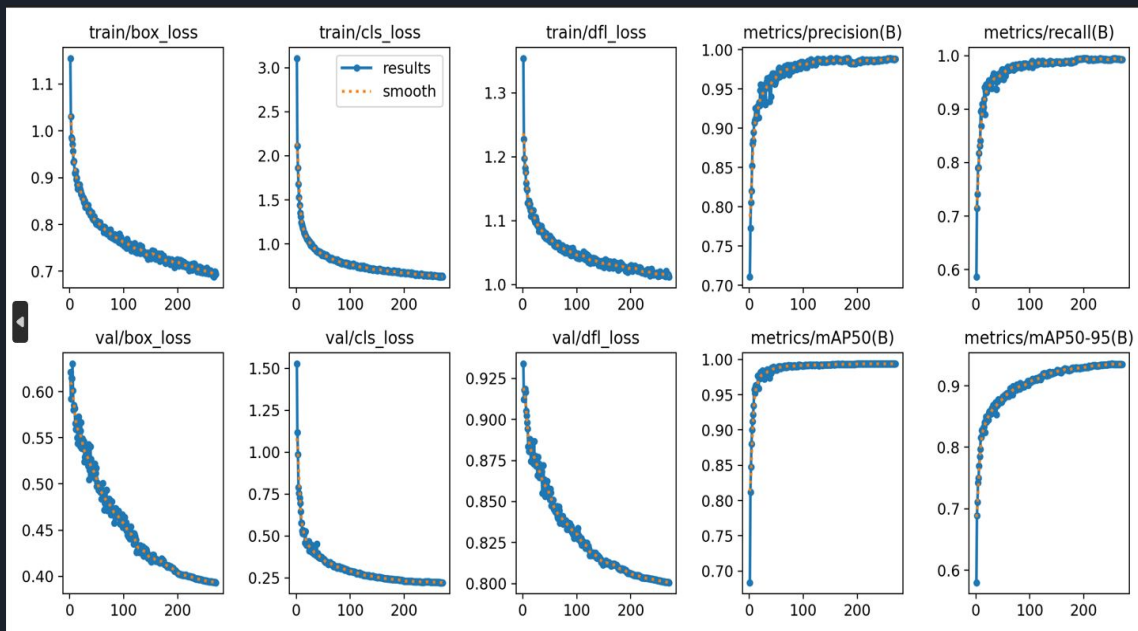
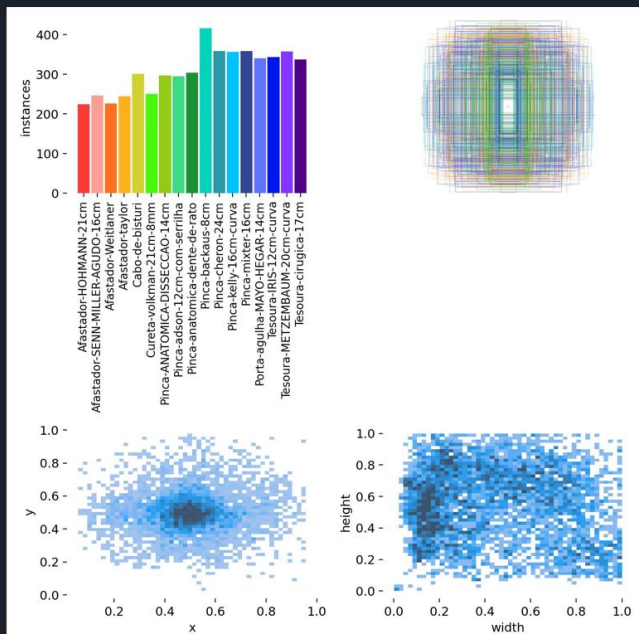
Arquivo PT

 **last.pt**

18/01/2024 19:07

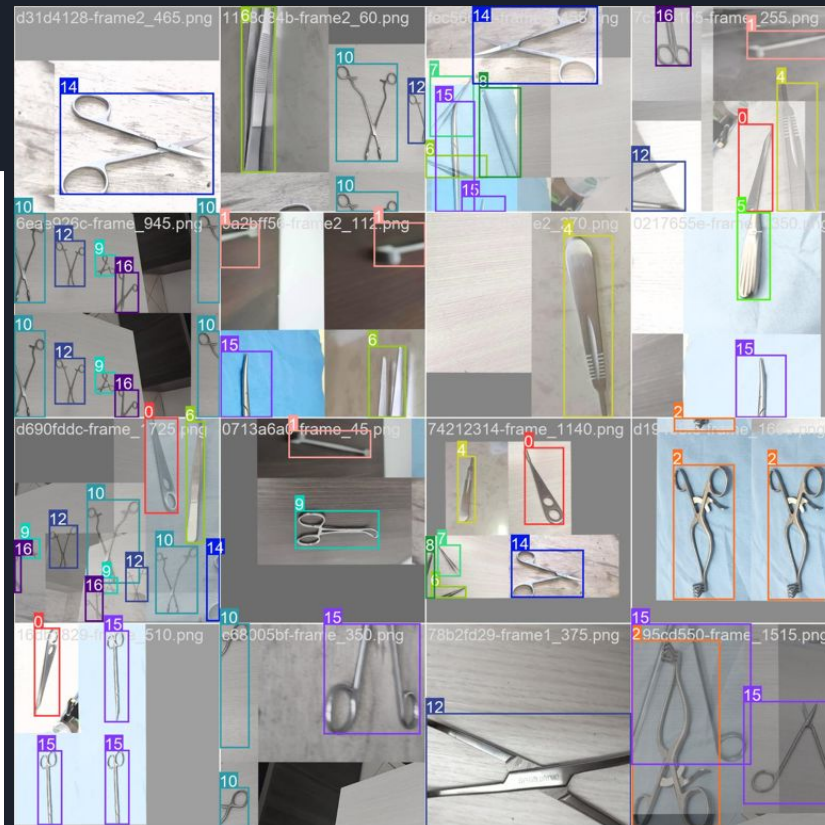
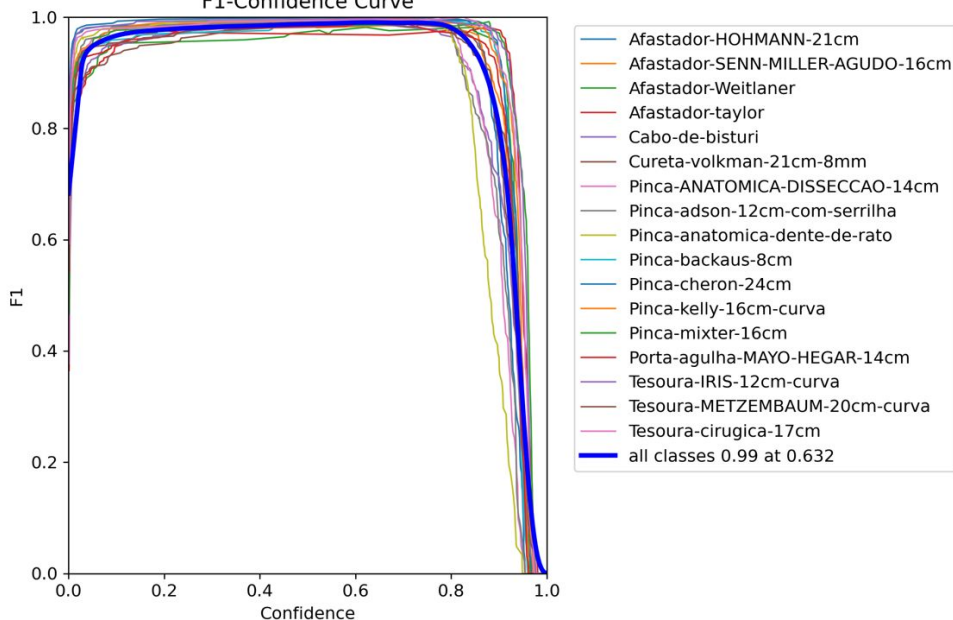
Arquivo PT

Informações do treinamento



Informações do treinamento

F1-Confidence Curve





Com os pesos é possível realizar as detecções de instrumentais

The image shows a surgical instrument segmentation interface on a computer monitor. The interface displays a top-down view of a surgical table with various instruments. Below the monitor, the same instruments are shown on a desk, each with a colored bounding box and a text label. The labels include the instrument name, dimensions, and a confidence score. The instruments are: a yellow box for a Hemostat (10.00), a green box for a Scissors (17.00), a blue box for a Forceps (17.00), an orange box for a Scissors (17.00), and a purple box for a Forceps (17.00). The labels are: "Hemostat 10.00", "Scissors 17.00", "Forceps 17.00", "Scissors 17.00", and "Forceps 17.00".

