**Testes realizados nos modelos de detecção: Yolo, Faster Rcnn e SSD.**

* Para fazer a detecção e rotulagem em 5000 imagens retiradas do nosso dataset para medir a eficiência de cada modelo.
* Foram consideradas as métricas de desempenho: Média de tempo gasta por imagem, Tempo total gasto e Total de imagens não detectadas.

Testes realizados no modelos pré-treinados na base de dados do Singapore Maritime Dataset:

**Modelos yolo:**

* [Full Yolo\_v2\_smd](https://mega.nz/)
* [Tiny\_Yolo\_v2\_smd](https://mega.nz/)
* [Squeezenet\_Yolo\_v2\_smd](https://mega.nz/)
* [Full\_Yolo\_v2\_smd\_2](https://mega.nz/)

Dataset 1 - O código extrai cada primeiro quadro de cada vídeo do SMD, com os primeiros 70% para treinamento e os 30% restantes para testes.

Dataset 2 - Mesma divisão, porém os quadros de 4 vídeos selecionados são adicionados completamente na parte de teste.

|  |  |  |  |
| --- | --- | --- | --- |
| **Dataset** |  | **Média de tempo gasta por imagem (s)** | **Tempo Total gasto em 5000 imagens (min)** |
| **1** | Full\_yolo\_v2 | 0.486 | 40.51 |
| tiny\_yolo\_v2 | 0.148 | 12.34 |
| squeezenet\_yolo\_v2 | 0.07510 | 6.25 |
| **2** | full\_yolo\_v2\_2 | 0.45 | 37.51 |

**Modelos Faster\_RCNN e SSD:**

* Faster\_rcnn\_inception\_v2
* Faster\_rcnn\_resnet101
* Faster\_rcnn\_resnet50
* Ssd\_inception\_v2
* Ssd\_mobilenet\_v1
* Ssd\_mobilenet\_v2
* Ssd\_resnet50\_v1
* Faster\_rcnn\_inception\_v2\_2
* Faster\_rcnn\_resnet50\_2
* Ssd\_mobilenet\_v1\_2
* **Threshold = 0.99**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Média de tempo gasta por imagem (s)** | **Tempo Total gasto em 5000 imagens (min)** | **Total de imagens não detectadas** |
| **Dataset 1** | **Faster\_rcnn\_inception\_v2** | **7.60** | **633.54** | **423** |
| **Faster\_rcnn\_resnet101** | **5.17** | **431.45** | **321** |
| **Faster\_rcnn\_resnet50** | **10.37** | **864.34** | **342** |
| **Ssd\_inception\_v2** | **2.53** | **211** | **3798** |
| **Ssd\_mobilenet\_v1** | **3.26** | **272.45** | **4625** |
| **Ssd\_mobilenet\_v2** | **2.20** | **183.91** | **4419** |
| **Ssd\_resnet50\_v1** | **6.70** | **558.76** | **4809** |
| **Dataset 2** | **Faster\_rcnn\_inception\_v2\_2** | **2.13** | **178.29** | **416** |
| **Faster\_rcnn\_resnet50\_2** | **3.44** | **287.41** | **275** |
| **Ssd\_mobilenet\_v1\_2** | **3.90** | **325.38** | **4560** |

* **Threshold = 0.4 (Dataset 1)**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Média de tempo gasta por imagem (s)** | **Tempo Total gasto em 5000 imagens (min)** | **Total de imagens não detectadas** |
| **Faster\_rcnn\_inception\_v2** | **8.69** | **724.82** | **19** |
| **Faster\_rcnn\_resnet101** | **9.99** | **832.8** | **12** |
| **Faster\_rcnn\_resnet50** | **3.46** | **288.89** | **15** |
| **Ssd\_inception\_v2** | **3.34** | **278** | **553** |
| **Ssd\_mobilenet\_v1** | **2.99** | **249.59** | **88** |
| **Ssd\_mobilenet\_v2** | **1.33** | **110.95** | **1009** |
| **Ssd\_resnet50\_v1** | **7.14** | **595.71** | **183** |

* **Threshold = 0.1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **Média de tempo gasta por imagem (s)** | **Tempo Total gasto em 5000 imagens (min)** | **Total de imagens não detectadas** | **Acurácia (%)** |
| **Dataset 1** | **Faster\_rcnn\_inception\_v2** | **2.05** | **170.84** | **4** | **56.18%** |
| **Faster\_rcnn\_resnet101** | **4.80** | **400.75** | **0** | **68.92%** |
| **Faster\_rcnn\_resnet50** | **3.47** | **289.68** | **2** | **79.12%** |
| **Ssd\_inception\_v2** | **2.26** | **189.01** | **80** | **86.34%** |
| **Ssd\_mobilenet\_v1** | **2.89** | **241.07** | **0** |  |
| **Ssd\_mobilenet\_v2** | **1.30** | **108.77** | **253** | **80.0%** |
| **Ssd\_resnet50\_v1** | **4.64** | **387.45** | **0** |  |
| **Dataset 2** | **Faster\_rcnn\_inception\_v2\_2** | **2.16** | **180.62** | **5** | **61.54%** |
| **Faster\_rcnn\_resnet50\_2** | **3.56** | **297.40** | **1** | **51.24%** |
| **Ssd\_mobilenet\_v1\_2** | **2.89** | **241.07** | **0** |  |

**\*Acurácia obtida através de análise e descarte das imagens rotuladas erroneamente**

* **Threshold = 0.6 (Dataset 1)**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Média de tempo gasta por imagem (s)** | **Tempo Total gasto em 5000 imagens (min)** | **Total de imagens não detectadas** |
| **Faster\_rcnn\_inception\_v2** | **2.99** | **249.59** | **41** |
| **Faster\_rcnn\_resnet101** |  |  |  |
| **Faster\_rcnn\_resnet50** | **3.50** | **291.83** | **25** |
| **Ssd\_inception\_v2** | **3.34** | **278.39** | **939** |
| **Ssd\_mobilenet\_v1** |  |  |  |
| **Ssd\_mobilenet\_v2** | **1.23** | **103.10** | **1660** |
| **Ssd\_resnet50\_v1** |  |  |  |

**Modelos de Segmentação**

* Os modelos Mask R\_CNN e PointRend\_Resnet50 são pré-treinados no conjunto de dados Microsoft Coco, com 80 categorias de objetos comuns.

Teste:

* O objetivo do teste foi encontrar outras possíveis alternativas para a rotulagem semi-automática, que possam ter melhor desempenho e fazermos uma comparação entre elas.
* Testes realizados nas mesmas 5000 imagens dos modelos de Singapura apresentados na reunião passada.
* Com o parâmetro confidence = 0.7.
* Apenas para a classe de barcos.
* A acurácia foi determinada pelo mesmo método que os modelos de Singapura.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Média de tempo gasto por imagem (s)** | **Tempo Total gasto em 5000 imagens (min)** | **Acurácia** |
| **Mask\_rcnn** | **0.75** | **62.55** | **96,2%** |
| **Pointrend\_resnet50** | **2.78** | **231.98** | **93.24%** |

**Notebooks no github:**

[Sinagapore-Maritime](https://github.com/beatrizmaraujo/MAI-DAI/tree/main/Sinagapore-Maritime)

**Generico\_Singapura.ipynb:** notebook que roda os modelos de Singapura, salva as imagens com as detecções e salva os .txt com as coordenadas da localização do objeto. Ele também exclui os arquivos txt sem nenhuma detecção (vazios) e salva as imagens sem detecção em outra pasta, para controle.

**Gray\_Exclude.ipynb:** Move as imagens em preto e branco de uma pasta para outra. Alguns modelos não rodam em imagens preto e branco, por isso separá-las.

**Video\_to\_photo.ipynb:** Gera frames de vídeos como imagens jpg.

[Segmentation](https://github.com/beatrizmaraujo/MAI-DAI/tree/main/Segmentation)

**Segmentation.ipynb:** notebook que roda os modelos de segmentação utilizando a biblioteca pixellib

[data-convertion](https://github.com/beatrizmaraujo/MAI-DAI/tree/main/data-convertion)

Nesta pasta se encontram vários notebooks úteis para conversão das anotações para outros formatos.