Documentação do projeto:

Desafio - Processo seletivo SmartAgri

Autor: Juan Pedro Baena Cassal



São José dos Campos, Sp

Maio – 2022

Sumário

[INTRODUÇÃO 4](#_Toc103085329)

[1 DEFINIÇÃO DA METODOLOGIA 5](#_Toc103085330)

[1.1 Cálculo do Desempenho 5](#_Toc103085331)

[2 PREPARAÇÃO DO AMBIENTE DE DESENVOLVIMENTO 7](#_Toc103085332)

[2.1 OpenCv 7](#_Toc103085333)

[2.2 Sys 7](#_Toc103085334)

[2.3 Random 7](#_Toc103085335)

[2.4 Pandas 8](#_Toc103085336)

[3 ELABORAÇÃO DO ALGORITMO 9](#_Toc103085337)

[4 TESTES PARA A ESCOLHA DO ALGORITMO 10](#_Toc103085338)

[5 ANÁLISE DOS RESULTADOS 11](#_Toc103085339)

Índice de Ilustrações

[Figura 1 - Modelo de tabela de resultados 4](#_Toc103085425)

[Figura 2 - Fluxograma de desenvolvimento 4](file:///C:\Users\juanp\PycharmProjects\pythonProject\Documentação%20do%20projeto.docx#_Toc103085426)

[Figura 3 - Fluxograma Macro da lógica do programa de rastreamento 9](file:///C:\Users\juanp\PycharmProjects\pythonProject\Documentação%20do%20projeto.docx#_Toc103085427)

[Figura 4 - Rastreamento de 5 pontos em uma máquina pulverizadora. 11](#_Toc103085428)

# INTRODUÇÃO

O desafio consiste em escrever um script em Python que analise um vídeo contendo a vista aérea (capturada com um drone) de uma máquina pulverizadora. O objetivo é utilizar visão computacional para localizar 5 pontos específicos na máquina para cada frame. O script deverá gerar um arquivo CSV contendo as coordenadas em pixel de cada ponto, em um formato de tabela especificado na figura 1.

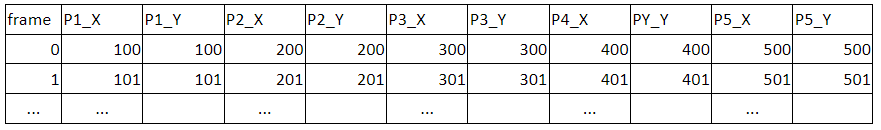


Figura 1 - Modelo de tabela de resultados

A execução do desafio conteve as seguintes etapas citadas no fluxograma a baixo na figura 2:

Definição da metodologia

Elaboração da Documentação

Preparação do ambiente de desenvolvimento

Elaboração do algoritmo

Testes para a escolha do algoritmo

Figura 2 - Fluxograma de desenvolvimento

# 1 DEFINIÇÃO DA METODOLOGIA

Para a realização do desafio a melhor ferramenta operacional para interpretar o vídeo é a visão computacional, um dos seguimentos da inteligência artificial (machine learning) que estuda o processamento de imagens do mundo real por um computador, sendo um vídeo a combinação de um conjunto de imagens (frames). Esta técnica, desenvolve teoria e tecnologia para a construção de sistemas artificiais que obtém informação de imagens ou quaisquer dados multidimensionais.

Dentro do seguimento de visão computacional existem ainda as técnicas de Detecção de objetos e o rastreamento, sendo a primeira utilizada para reconhecer o objeto em uma imagem ou vídeo e a segunda para a detecção continua do tempo do objeto em vídeo. Para este trabalho será utilizado o modelo de reconhecimento dos objetos no frame inicial e a partir dele será feito o rastreamento dos objetos.

Para o rastreamento dos objetos existem algoritmos padrões consolidados no machine learning e amplamente utilizados na visão computacional, este projeto irá utilizar como base para testes os algoritmos citados a baixo, até que um deles seja escolhido como o que melhor atende o objetivo do projeto na fase de testes.

* BOOSTING;
* MIL;
* KCF;
* TLD;
* MEDIANFLOW;
* MOSSE;
* CSRT.

Em teoria, o algoritmo BOOSTING é o de menor precisão, entretanto, o mais rápido e o algoritmo CSRT, o mais preciso, mas também, o mais lento.

* 1. Cálculo do Desempenho

É importante medir o quanto eficiente é o programa em encontrar os resultados comparado com o consumo de processamento da máquina, pois caso o processamento for insustentável, o algoritmo não conseguirá cumprir o seu objetivo de rastreamento, o que o tornará pouco eficiente. Dessa forma, para medir a eficiência do programa, será utilizado um cálculo baseado no fps – frames por segundo, ou seja, o quanto o programa consegue ser preciso por frames por segundo. Os detalhes da teoria do cálculo de fps pode ser encontrado no endereço web a baixo:

<https://docs.opencv.org/4.x/dc/d71/tutorial_py_optimization.html>

1. PREPARAÇÃO DO AMBIENTE DE DESENVOLVIMENTO

Para desenvolver o projeto na linguagem de programação Python e utilizando bibliotecas da mesma linguagem é preciso criar um ambiente para o projeto aonde serão salvas todas as bibliotecas utilizadas, pois caso seja necessário reproduzir o programa no futuro se caso o usuário faça download das bibliotecas em sua versão mais atual, pode haver incompatibilidades que impossibilitem a execução.

O Ambiente utilizado para o projeto foi em Conda, o qual utiliza a Anaconda3, uma distribuição das linguagens de programação Python e R para computação científica, que visa simplificar o gerenciamento e implantação de pacotes. Através do conda é possível visualizar todas as bibliotecas instaladas no ambiente de forma gráfica pelo Anaconda Navigator.

Para realizar o download das bibliotecas utilizadas no projeto será utilizado o pip, um sistema de gerenciamento de pacotes padrão de facto usados para instalar e gerenciar pacotes de software escritos em Python.

Para realização do projeto foi utilizado 3 bibliotecas básicas:

* 1. OpenCv

A OpenCv, citado pelo programa como cv2, é uma biblioteca multiplataforma, totalmente livre ao uso acadêmico e comercial, para o desenvolvimento de aplicativos na área de Visão computacional. Este projeto utiliza a versão 4.4.5.56.

* 1. Sys

Sys é uma biblioteca com diversas funções voltadas principalmente para trabalhar com as "configurações" obtidas na execução atual de um script, por exemplo, pegar parâmetros, pegar variáveis de ambiente, pegar caminho do script ou de onde está sendo executado, fora que proporciona algum controle na interface de linha de comando, o que pode variar pelo tipo de interface ou terminal.

* 1. Random

Este módulo implementa geradores de números pseudoaleatórios para várias distribuições, é utilizado no programa para gerar cores aleatórias para as bounding box, caixas delimitadoras do rastreamento.

* 1. Pandas

Pandas é uma biblioteca de software criada para a linguagem Python para manipulação e análise de dados. Em particular, oferece estruturas e operações para manipular tabelas numéricas e séries temporais. No programa é utilizada para manipular listas, transformando-as em dataframes, estrutura de dados bidimensional com os dados alinhados de forma tabular em linhas e colunas, para então, exportar os dados para um arquivo csv.

1. ELABORAÇÃO DO ALGORITMO

Para a elaboração do algoritmo será utilizada a linguagem e programação Python, uma linguagem de alto nível, interpretada de script, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte. A lógica do programa pode ser visualizada na figura a baixo através de um fluxograma macro.

importar bibliotecas

Definir o algoritmo base da visão computacional provindo da biblioteca OpenCv

Escolha do vídeo

Seleção dos objetos para o rastreamento

Rastreamento

Cálculo FPS de eficiência

Registro das coordenadas

Exportar resultados para CSV

Figura 3 - Fluxograma Macro da lógica do programa de rastreamento

Para o detalhamento do programa, verificar o código fonte contido na mesma pasta desta documentação, cada etapa do código foi comentada e explicada.

1. TESTES PARA A ESCOLHA DO ALGORITMO

O programa permite que o usuário escolha de forma manual o tipo de algoritmo de visão computacional será importado da biblioteca OpenCv e utilizado para o rastreamento, o que se torna necessário, uma vez que cada tipo de vídeo será um desafio singular e poderá exigir algoritmos de maiores precisões ou aqueles de menores já bastariam para atendê-los.

É importante mencionar que na biblioteca openCv quanto maior a precisão do algoritmo maior será a lentidão da execução do programa, por isso para não prejudicar a eficiência do mesmo, calculada pelo FPS, caso o rastreamento consiga ser feito pelo algoritmo mais simples, ele deverá ser o escolhido para manter o FPS mais alto o possível.

Para este programa será escolhido o algoritmo chamado de BOOSTING, um método usado no Machine Learning para reduzir erros na análise preditiva de dados, apresenta alta performance preditiva dos modelos de máquina, convertendo vários aprendizes fracos em um único modelo de aprendizado forte. Este algoritmo conseguiu em 10 minutos de processamento rastrear os objetos de forma correta durante o vídeo e não houve necessidade de escolher outro com maior precisão.

# ANÁLISE DOS RESULTADOS

Através do algoritmo desenvolvido por este projeto consegue de forma objetiva realizar o rastreamento dos pontos propostos de uma máquina pulverizadora em um vídeo real filmado via drone.

Foi utilizado apenas técnicas de rastreamento de objetos, entretanto caso houvesse maiores dificuldades, poderia ter sido tomado dois caminhos, o primeiro seria a troca de algoritmo de Machine Learning para outro com maior precisão e caso a dificuldade permanecesse, poderia ter sido utilizado uma mistura no mesmo código de detecção e rastreamento, o que tornaria o código ainda mais confiável.

Foi exportado uma amostra contendo os frames iniciais dos vídeos e coordenadas para a análise em Excel.

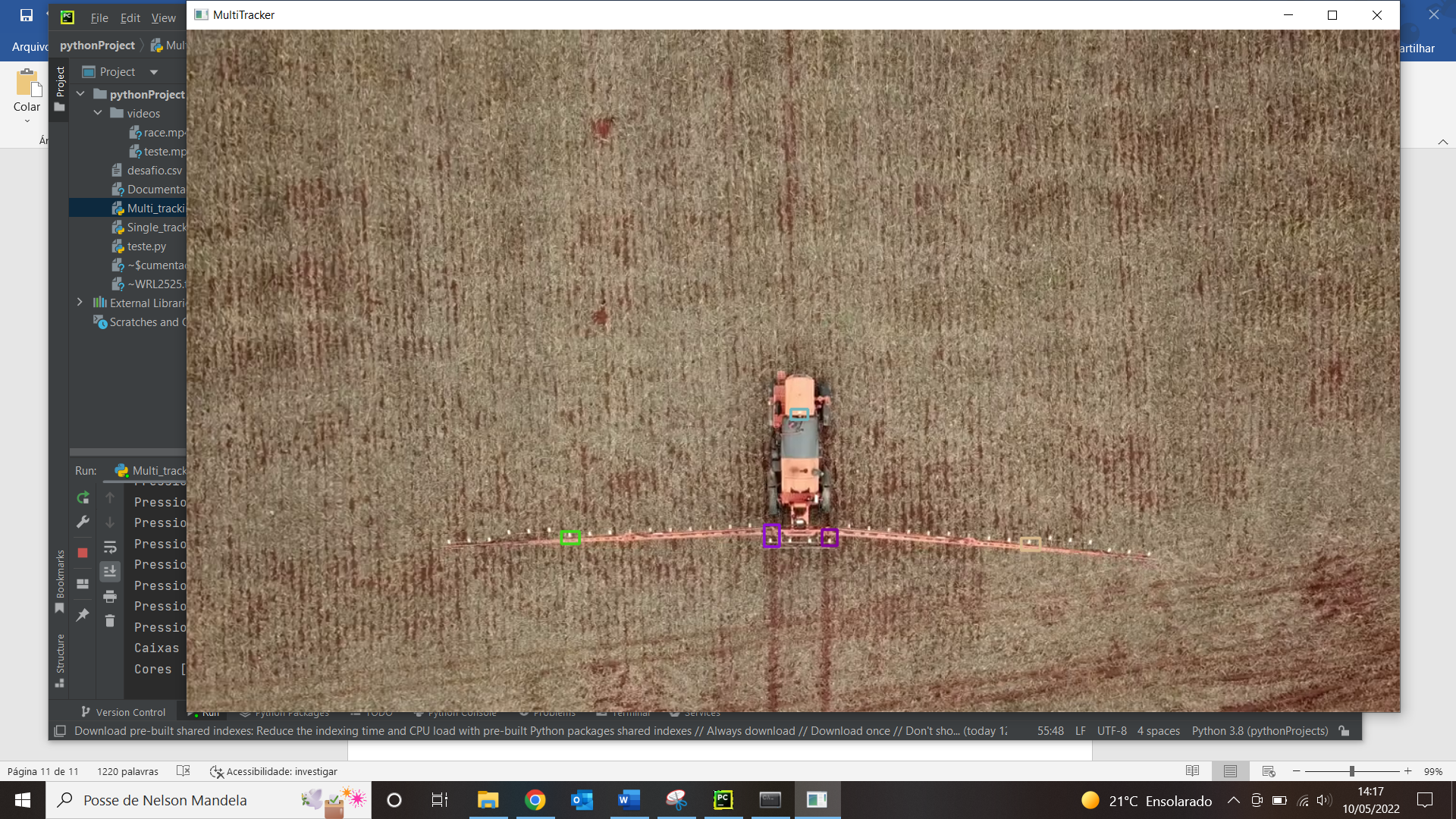


Figura 4 - Rastreamento de 5 pontos em uma máquina pulverizadora.