

Assuntos:

1. Conhecer o projeto
2. Lista de material
3. Lista de softwares
4. Demonstração de funcionamento
5. Projeto com base em pacotes: OpenCV, CVlib e Yolo

Preparando o ambiente Windows

FIAP



Visual Studio Code



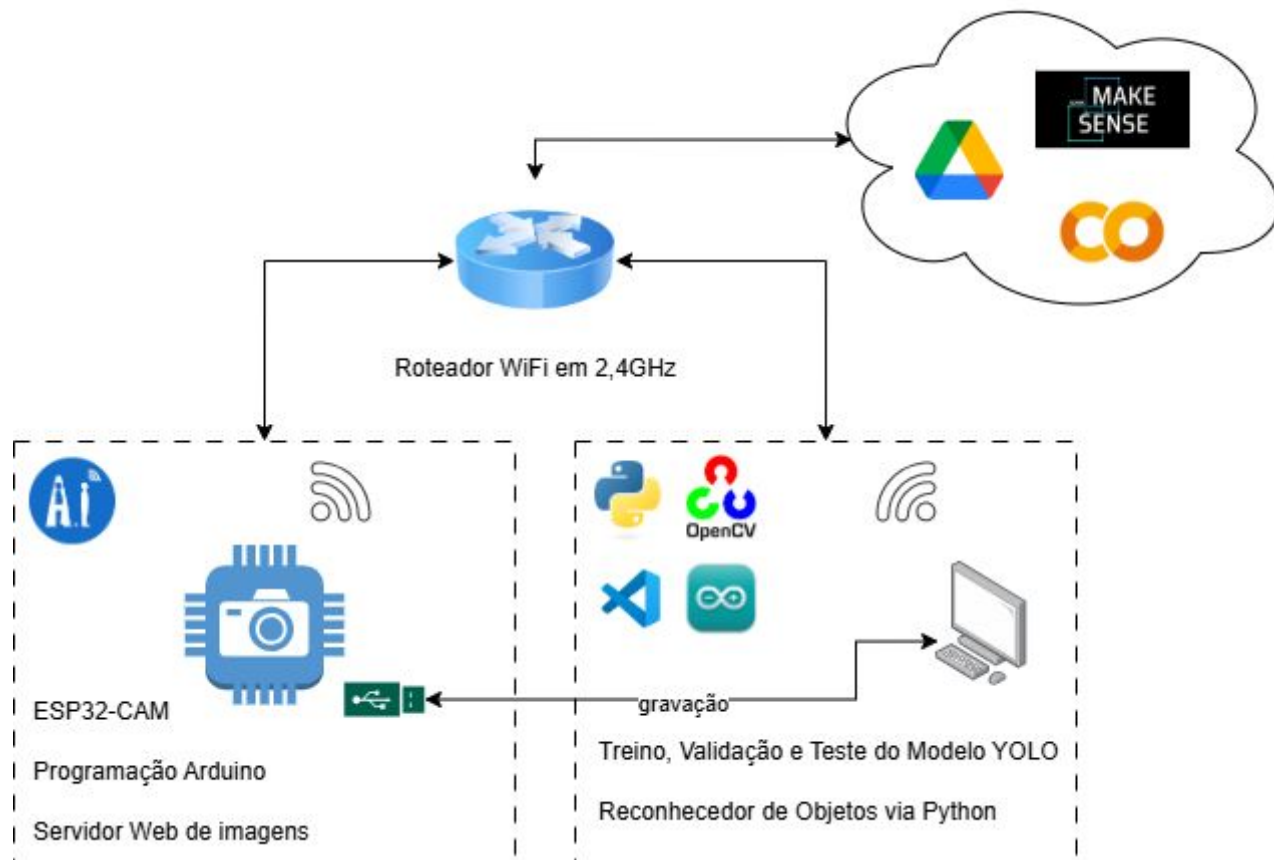
python™



Você vai dos seguintes itens:

- 1) Arduino IDE (a mais recente funciona, porém recomendo a Legacy 1.8.19) →
 - a) Instalar a biblioteca da família ESP
- 2) Python 3 (versão atual) e instalar as seguintes bibliotecas e suas versões:
 - a) opencv-python → versão atual
 - b) cvlib → versão atual → dataset Yolo ([lista de 80 objetos](#) que vai reconhecer)
 - c) matplotlib → versão atual
 - d) tensorflow → 2.16.1
- 3) Visual Studio Code Microsoft (a mais recente funciona)
- 4) Comprar um ESP32-CAM
- 5) Conectar eletronicamente a placa ESP32-CAM com o módulo de comunicação FTDI ou Módulo Gravação
- 6) Se usar um módulo FTDI separado, vai precisar de 5 jumpers fêmea-fêmea de 15 ou 20cm. Veja o slide a seguir para entender melhor.

Arquitetura do projeto



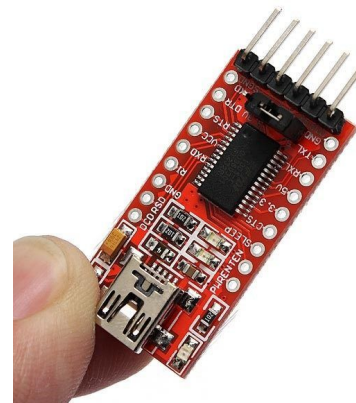
Lista de material



ESP32-CAM



5 jumper arduino
fêmea-fêmea de
15 ou 20cm

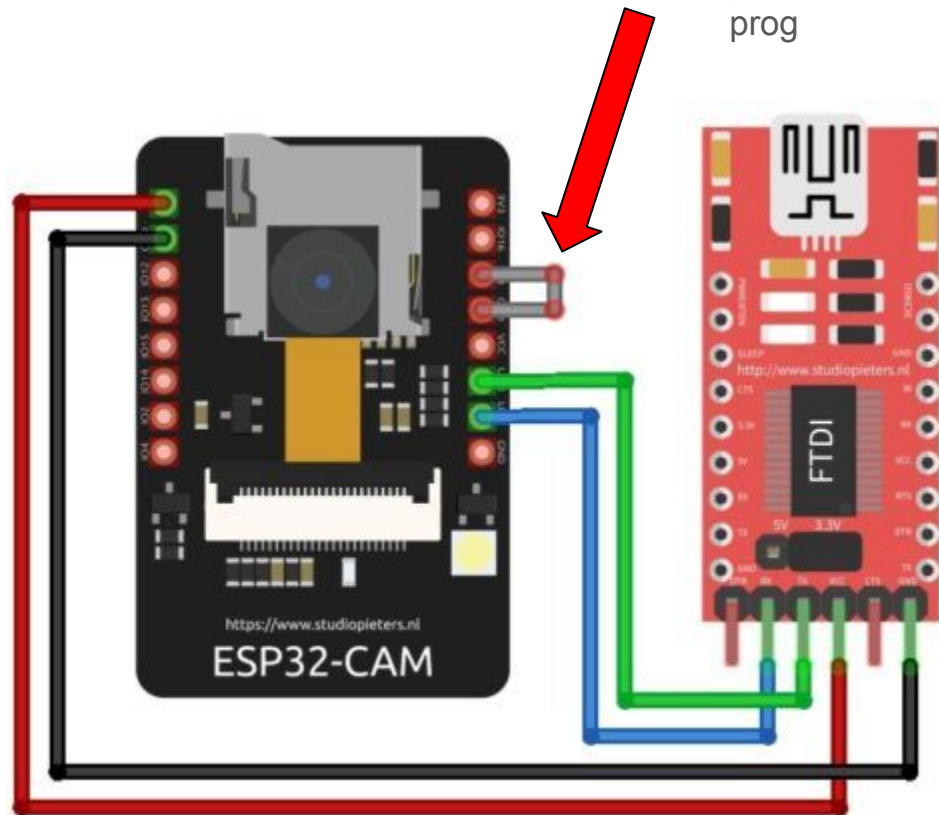


Usa cabo
mini USB
ou
USB-C

Usado para
descarregar o
código no
ESP32-CAM
usando USB

Conectando o ESP32-CAM com o módulo FTDI

Você precisa colocar esse jumper e pressionar o RESET para entrar no modo prog



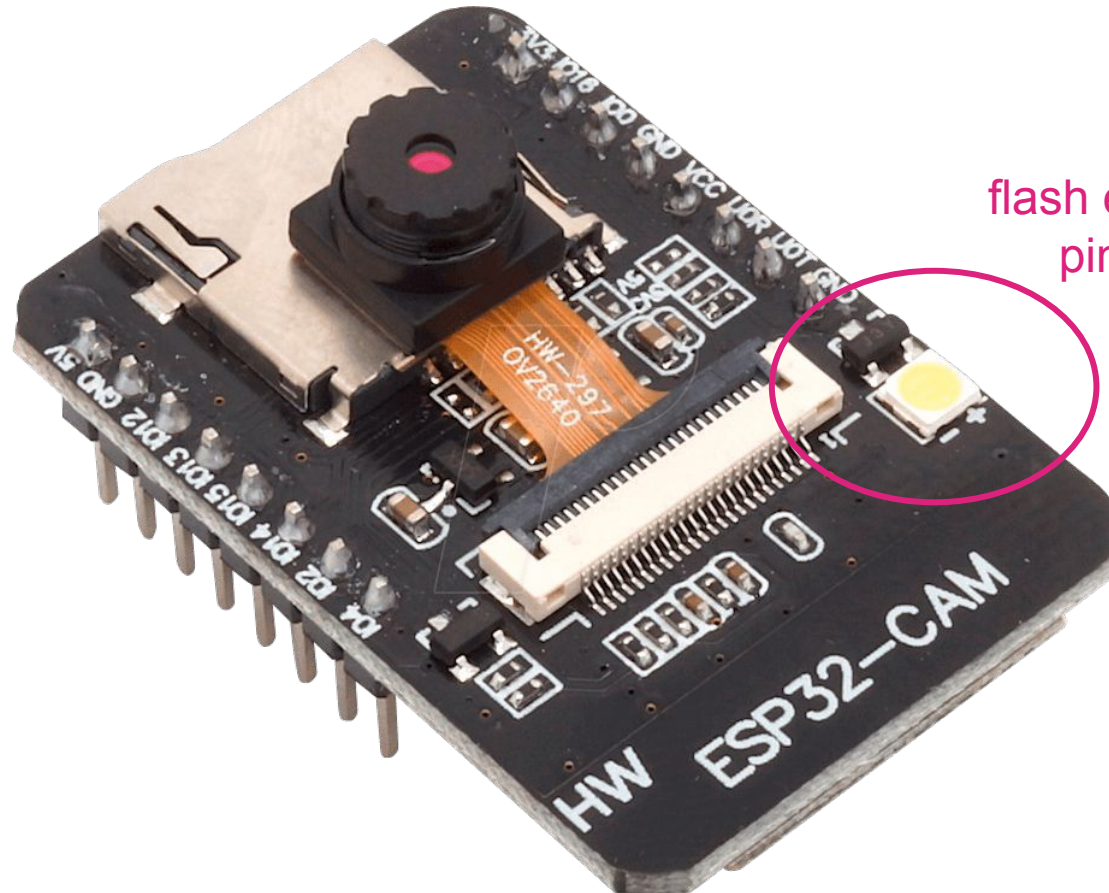
Depois que grava [100%], vc remove o jumper, pressione RESET novamente.

Caso necessite, abra a lupa no MONITOR SERIAL do Arduino IDE para acompanhar as msg de controle.

Conectando o ESP32-CAM com o módulo FTDI

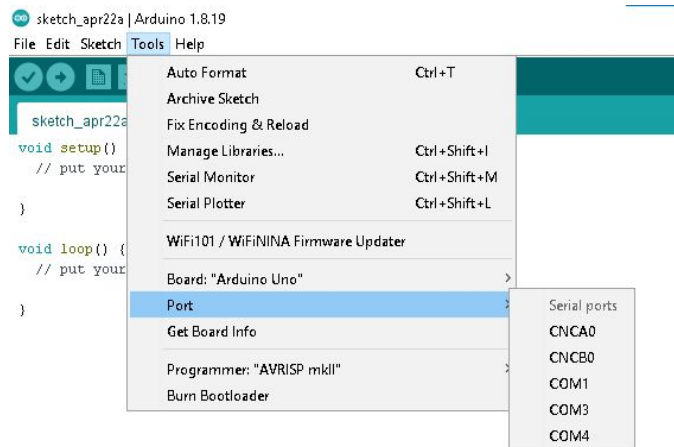
Pinos do ESP32-CAM	Pinos Placa FTDI	Descrição	Cores
5V	VCC	Alimentação positiva	Roxo ou Vermelho
GND	GND	Alimentação negativa	Preto ou Marrom
UOT	RX	Pinos de comunicação cruzados. TX ligado no RX	Azul ou Verde
UOR	TX	Pinos de comunicação cruzados. RX ligado no TX	Branco ou Amarelo
IO0 com GND	-----	Habilita o modo prog.	Cinza ou Laranja

Conectando o ESP32-CAM com o módulo FTDI



Testando o seu kit

- 1) Abra o Arduino IDE no seu computador;
- 2) Conecte o cabo USB na porta USB do seu computador;
- 3) Descubra qual porta COM a sua placa FTDI está alocada. Como fazer isso?
 - a) Vá em TOOLS, PORT e memorize quais as portas estão listadas com o cabo USB conectado;
 - b) Remova o cabo USB do computador e atualize a lista do PORT, reabrindo as telinhas do Arduino IDE.
 - c) Geralmente é a COM5



Se você estiver usando Notebook pessoal, execute essa tela



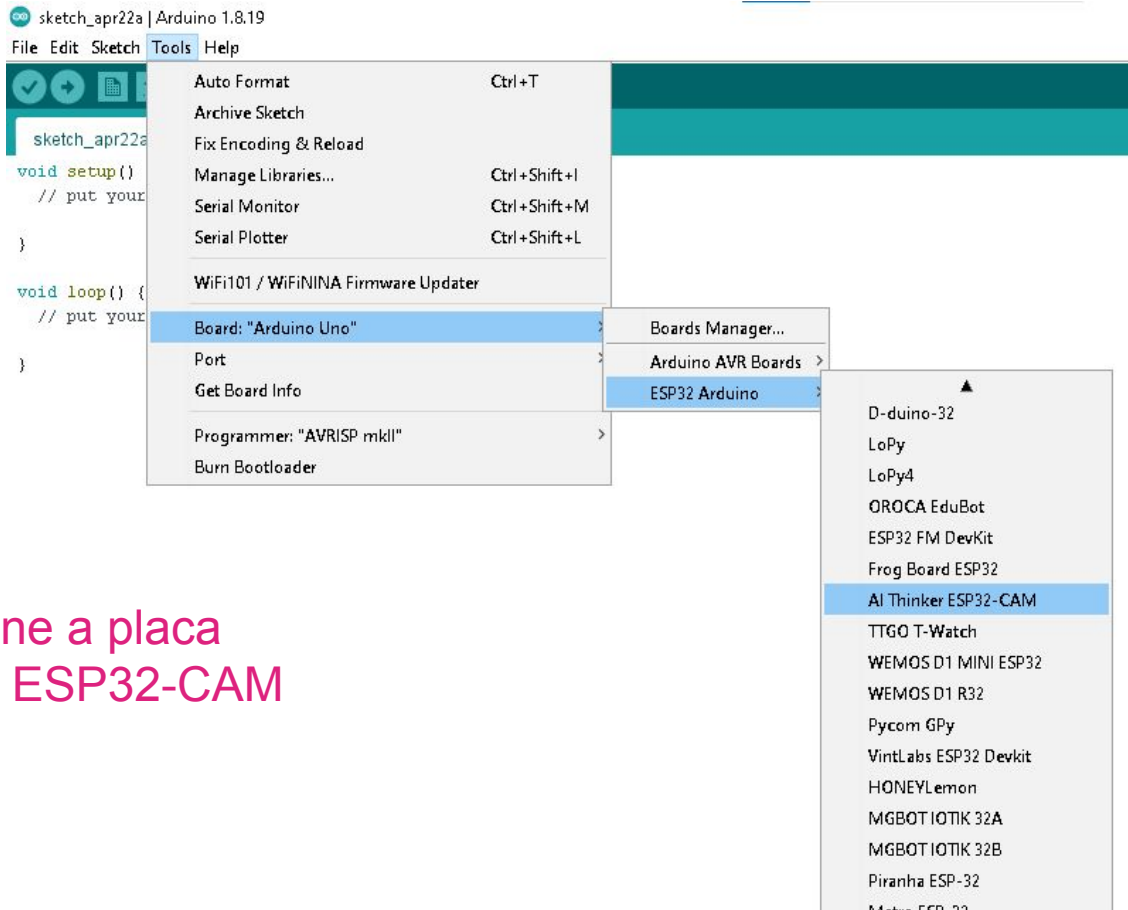
- 1) [caso esteja no PC da FIAP, pule esse item] Abra o Arduino IDE, e clique em ARQUIVO, depois PREFERÊNCIAS, e no campo URL ADICIONAIS PARA GERENCIADORES DE PLACAS cole esse endereço
https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
- 2) [caso esteja no PC da FIAP, pule esse item] Clique em OK
- 3) No menu horizontal, vá em FERRAMENTAS, depois vá em PLACA, vá em GERENCIADOR DE PLACAS, busque por ESP32 em boards, e pegue a última POR ESPRESSIF SYSTEMS [instalar]
- 4) No menu horizontal, vá em FERRAMENTAS, depois vá em PLACA, depois ESP32 ARDUINO e escolha a placa AI THINKER ESP32-CAM

4) Grave esse código do pisca-pisca

```
void setup() {  
    pinMode(4, OUTPUT);  
}  
  
void loop() {  
    digitalWrite(4, HIGH); // turn the LED on (HIGH is the voltage level)  
    delay(100);            // wait for a second  
    digitalWrite(4, LOW);  // turn the LED off by making the voltage LOW  
    delay(100);            // wait for a second  
}
```

- 5) Programe um pisca-pisca no Arduino IDE e considere o delay de 100ms [atenção, flash muito tempo no HIGH com delay de > 1000 ms pode queimá-lo]
- 6) Selecione a placa conforme o print a seguir

Testando o seu kit



Selecione a placa
AI Thinker ESP32-CAM

- 4) Como gravar o código-fonte na placa?
 - a) Com o modelo da placa selecionado para AI Thinker ESP32-CAM
 - b) Com a porta COM correta
 - c) Com o fio jumper conectado entre IO0 no GND
 - d) Clique no botão SETA DIREITA
 - e) Se aparecer na tela, clique num botão que tem na plaquinha. Esse clique tem que acontecer enquanto os pontinhos estão sendo impressos
 - f) Após 100% de upload, puxe o fiozinho e pressione o botão reset da placa novamente

A screenshot of the Arduino IDE interface. The title bar shows 'sketch_apr22a | Arduino 1.8.19'. The menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. The toolbar contains several icons, with the 'Upload' button (a right-pointing arrow) circled in pink. Below the toolbar, the code editor shows the following code:

```
sketch_apr22a $  
void setup() {  
    // put your setup code here, to run once:  
  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
  
}
```

- 1) Se você não possui o Arduino IDE, clique [AQUI](#) e baixe a versão compatível com o seu sistema operacional;
- 2) Após ter instalado com sucesso o Arduino IDE, abra-o, e clique em ARQUIVO, depois PREFERÊNCIAS, e no campo URL ADICIONAIS PARA GERENCIADORES DE PLACAS cole esse endereço
https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
- 3) Clique em OK, DEPOIS vá em FERRAMENTAS / PLACAS e clique em GERENCIAR PLACAS. Busque por ESP32 e instale o pacote de placas;
- 4) No menu horizontal, vá em FERRAMENTAS, depois vá em PLACA, depois ESP32 ARDUINO e escolha a placa AI THINKER ESP32-CAM
- 5) Baixe esse [código-fonte](#) INO (ou o copie e cole no sketch vazio que o Arduino cria como padrão) e baixe as [bibliotecas](#) necessárias. Adicione essas 3 bibliotecas seguindo esse caminho de menu da sua IDE: SKETCH → INCLUDE LIBRARY → ADD .ZIP LIBRARY

Testando o Reconhecedor de Objetos

sketch_apr22a | Arduino 1.8.19

File Edit Sketch Tools Help



sketch_apr22a \$

```
#include <WebServer.h>
#include <WiFi.h>
#include <esp32cam.h>
```



Acesse o monitor serial
do Arduino IDE clicando
na lupa

O monitor serial serve para você visualizar os comandos
Serial.println()

Testando o Reconhecedor de Objetos



```
COM11

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0030,len:1344
load:0x40078000,len:13864
load:0x40080400,len:3608
entry 0x400805f0

CAMERA OK
http://192.168.3.113
/cam-lo.jpg
/cam-hi.jpg
/cam-mid.jpg

☒ Auto-rolagem ☐ Show timestamp
Nova-linha 115200 velocidade Deleta a saída
```

Confira seu baud rate do
MONITOR SERIAL. Ele
tem que estar em 115200
bps

Essa mensagem indica
sucesso na gravação do
código-fonte do reconhecedor
de objetos

Para visualizar a imagem capturada pela sua câmera, faça:

- a) Abra o navegador de sua preferência. No Google Chrome funciona.
- b) Digite na barra URL: `http:// + IP que o seu monitor Serial mostra + /cam-hi.jpg`
- c) Dê um ENTER
- d) Deve aparecer uma foto na sua frente

Fazendo Download do Python no Windows

[Baixe aqui](#)

Preste atenção na versão indicada.

Use o Ctrl + f para buscar o 3.10.6

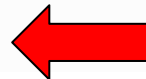
Outras versões podem funcionar, mas não fiz testes. Sugiro não perder tempo com outras versões. Use a versão indicada



▪ Python 3.10.6 - Aug. 2, 2022

Note that Python 3.10.6 cannot be used on Windows 7 or earlier.

- Download [Windows embeddable package \(32-bit\)](#)
- Download [Windows embeddable package \(64-bit\)](#)
- Download [Windows help file](#)
- Download [Windows installer \(32-bit\)](#)
- Download [Windows installer \(64-bit\)](#)



Instalação do Python no Windows

Preste atenção na tela inicial e marque **Add Python to Path** e siga com **Install Now**

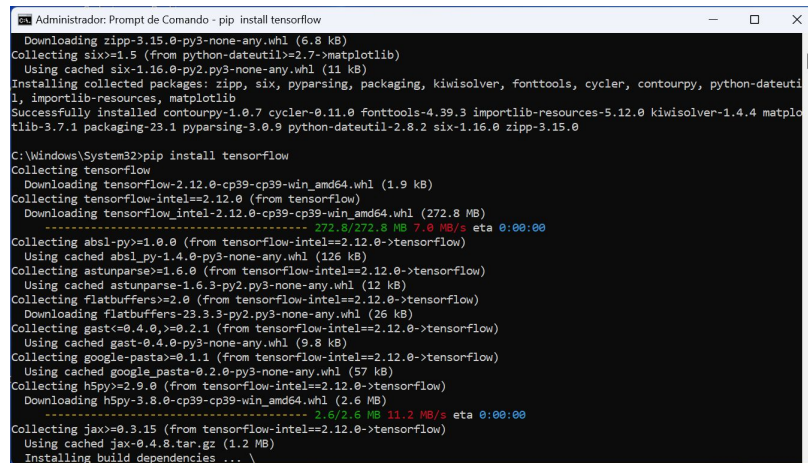


Instalação do Python no Windows - Instalando Bibliotecas

Vá no menu do Windows, abra o CMD do Windows no modo ADMINISTRADOR (use o botão direito do mouse sobre o CMD). Após aberto, digite:

- 1) `python.exe -m pip install --upgrade pip` ou `-m pip install --upgrade pip`
- 2) `pip install opencv-python`
- 3) `pip install cvlib`
- 4) `pip install matplotlib`
- 5) `pip install tensorflow==2.16.1`

para conferir as versões que vc tem faça:
`pip freeze` ou `pip list -o`
`python -V` → vc descobre o seu Python
(pesquise mais sobre esses comandos)



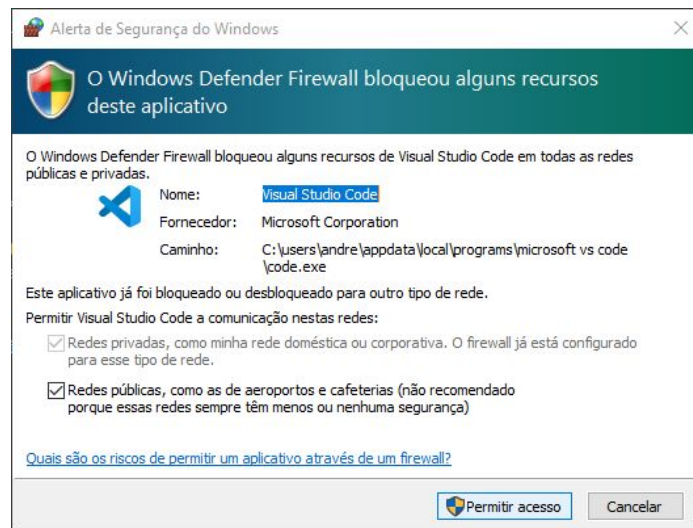
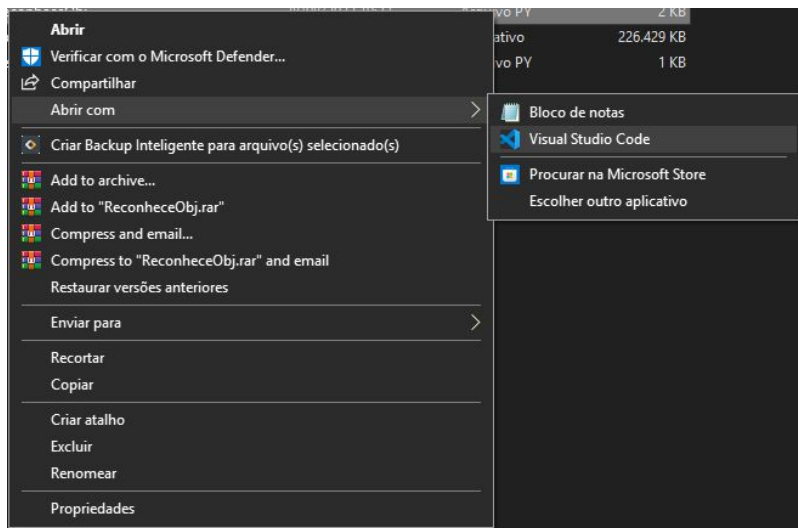
```
Administrador: Prompt de Comando - pip install tensorflow
Downloading zipp-3.15.0-py3-none-any.whl (6.8 kB)
Collecting six>=1.5 (from python-dateutil>=2.7->matplotlib)
Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: zipp, six, packaging, kiwisolver, fonttools, cycler, contourpy, python-dateutil, importlib-resources, matplotlib
Successfully installed contourpy-1.0.7 cycler-0.11.0 fonttools-4.39.3 importlib-resources-5.12.0 kiwisolver-1.4.4 matplotlib-3.7.1 packaging-23.1 pyparsing-3.0.9 python-dateutil-2.8.2 six-1.16.0 zipp-3.15.0

C:\Windows\System32>pip install tensorflow
Collecting tensorflow
  Downloading tensorflow-2.12.0-cp39-cp39-win_amd64.whl (1.9 kB)
Collecting tensorflow-intel==2.12.0 (from tensorflow)
  Downloading tensorflow_intel-2.12.0-cp39-cp39-win_amd64.whl (272.8 MB)
----- 272.8/272.8 MB 7.0 MB/s eta 0:00:00
Collecting absl-py>=1.0.0 (from tensorflow-intel==2.12.0->tensorflow)
Using cached absl_py-1.4.0-py3-none-any.whl (126 kB)
Collecting astunparse>=1.6.0 (from tensorflow-intel==2.12.0->tensorflow)
Using cached astunparse-1.6.3-py2.py3-none-any.whl (12 kB)
Collecting flatbuffers>=2.0 (from tensorflow-intel==2.12.0->tensorflow)
  Downloading flatbuffers-23.3.3-py2.py3-none-any.whl (26 kB)
Collecting gast<=0.4.0,>=0.2.1 (from tensorflow-intel==2.12.0->tensorflow)
Using cached gast-0.4.0-py3-none-any.whl (9.8 kB)
Collecting google-pasta>=0.1.1 (from tensorflow-intel==2.12.0->tensorflow)
Using cached google_pasta-0.2.0-py3-none-any.whl (57 kB)
Collecting h5py>=2.9.0 (from tensorflow-intel==2.12.0->tensorflow)
  Downloading h5py-3.8.0-cp39-cp39-win_amd64.whl (2.6 MB)
----- 2.6/2.6 MB 11.2 MB/s eta 0:00:00
Collecting jax>=0.3.15 (from tensorflow-intel==2.12.0->tensorflow)
Using cached jax-0.4.8.tar.gz (1.2 MB)
Installing build dependencies ... \
```

- 1) Baixe o instalador [AQUI](#)
- 2) E instale seguindo as opções básicas que aparecerem na sua tela até finalizar tudo.
- 3) Quando você abrir o programa pela 1ª vez, ele deve pedir para atualizar algumas coisas e então, você pode confirmar que sim.

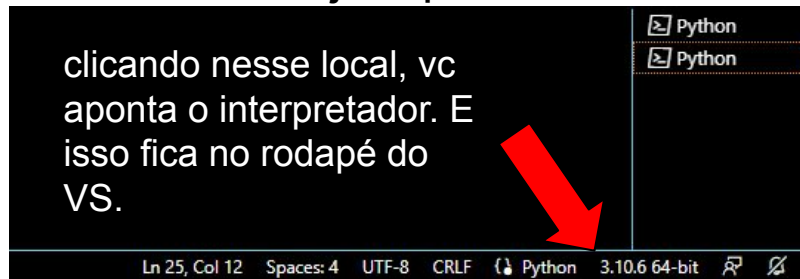
Como rodar o programa?

- 1) Baixe o código-fonte do Reconhecedor de Objetos [AQUI](#) e salve em diretório de forma organizada;
- 2) Clique no arquivo ReconheceObj.py com o botão direito do mouse e abra-o com o Visual Studio Code;
- 3) Caso apareça a tela de Alerta de Segurança, clique em PERMITIR ACESSO



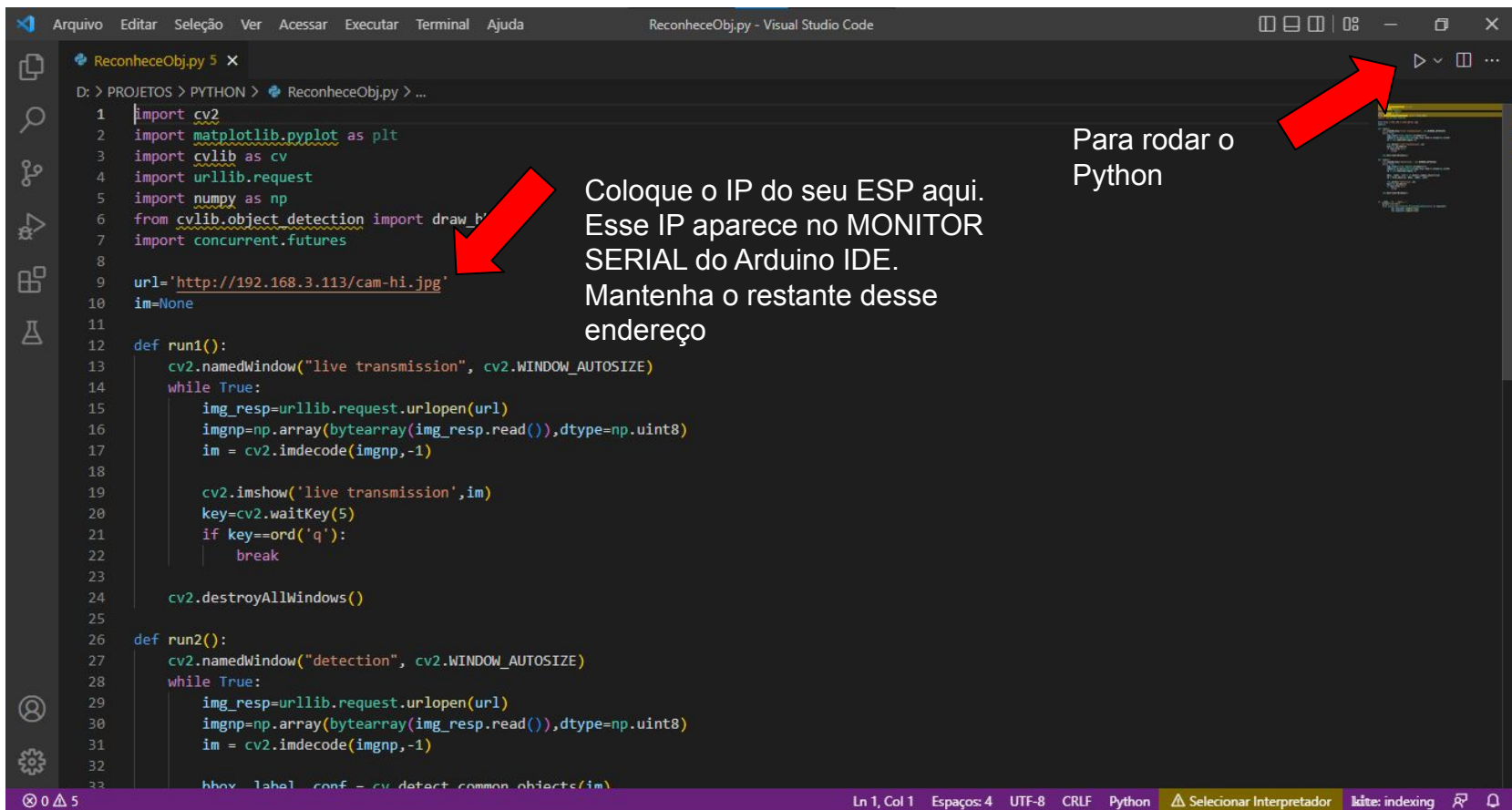
Como rodar o programa?

- 4) Quando abrir o [ReconheceObj.py](#) pela 1ª vez, o Visual Studio vai te pedir para “amarrar” com um INTERPRETADOR VÁLIDO do Python com o Visual Studio. Confirme essa associação com o PYTHON QUE VOCÊ INSTALOU. Isso só é feito uma vez. Se o Visual Studio não te fez essa pergunta, vc pode amarrar o interpretador de forma manual. Veja o print abaixo



- 5) Pegue o IP da sua placa através do MONITOR SERIAL da IDE Arduino e edite a linha 10 do código [ReconheceObj.py](#) (veja o slide a seguir).
- 6) Clique no botão PLAY para rodar o Python no Visual Studio Code. **Veja o slide seguinte que mostra onde fica o PLAY.**

Como rodar o programa?

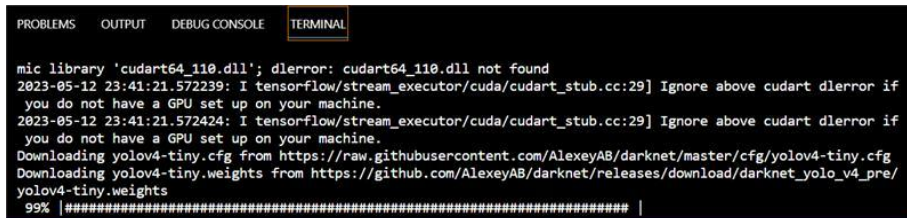


```
1 import cv2
2 import matplotlib.pyplot as plt
3 import cvlib as cv
4 import urllib.request
5 import numpy as np
6 from cvlib.object_detection import draw_h
7 import concurrent.futures
8
9 url='http://192.168.3.113/cam-hi.jpg'
10 im=None
11
12 def run1():
13     cv2.namedWindow("live transmission", cv2.WINDOW_AUTOSIZE)
14     while True:
15         img_resp=urllib.request.urlopen(url)
16         imgnp=np.array(bytearray(img_resp.read()),dtype=np.uint8)
17         im = cv2.imdecode(imgnp,-1)
18
19         cv2.imshow('live transmission',im)
20         key=cv2.waitKey(5)
21         if key==ord('q'):
22             break
23
24     cv2.destroyAllWindows()
25
26 def run2():
27     cv2.namedWindow("detection", cv2.WINDOW_AUTOSIZE)
28     while True:
29         img_resp=urllib.request.urlopen(url)
30         imgnp=np.array(bytearray(img_resp.read()),dtype=np.uint8)
31         im = cv2.imdecode(imgnp,-1)
32
33         bbox, label, conf = cv.detect_common_objects(im)
```

Coloque o IP do seu ESP aqui.
Esse IP aparece no MONITOR
SERIAL do Arduino IDE.
Mantenha o restante desse
endereço

Para rodar o
Python

- 1) A primeira vez que rodar o Reconhecedor de Objetos [AQUI](#) é normal não acontecer nada, pois significa que a biblioteca YOLO está sendo instalada;



```
mic library 'cudart64_110.dll'; dLError: cudart64_110.dll not found
2023-05-12 23:41:21.572239: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dLError if
you do not have a GPU set up on your machine.
2023-05-12 23:41:21.572424: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dLError if
you do not have a GPU set up on your machine.
Downloading yolov4-tiny.cfg from https://raw.githubusercontent.com/AlexeyAB/darknet/master/cfg/yolov4-tiny.cfg
Downloading yolov4-tiny.weights from https://github.com/AlexeyAB/darknet/releases/download/darknet_yolo_v4_pre/
yolov4-tiny.weights
99% |#####|
```

- 2) Se aparecer **Iniciado... Ignore mensagens de erro do <cudart>** no terminal do Visual Studio, significa que está indo tudo bem até agora.



```
ur machine.

Iniciado... Ignore mensagens de erro do <cudart>

2023-05-13 10:44:39.742199: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudart64_110.dll'; dle
rror: cudart64_110.dll not found
2023-05-13 10:44:39.742830: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dLError if you do not have a GPU set up on yo
ur machine.
```

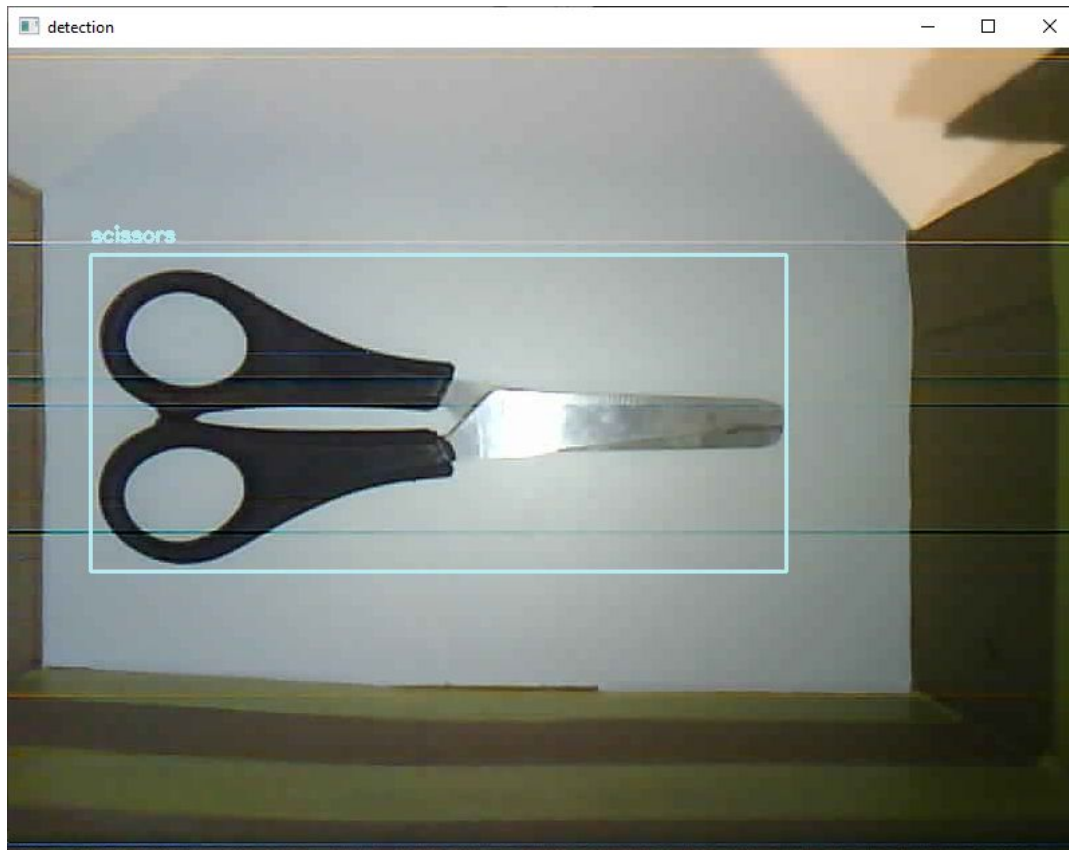

Resultado

Para encerrar o programa Python, pressione a tecla “q” em cada uma das telas selecionadas: Live e Detection.

O MONITOR SERIAL do Arduino IDE não precisa permanecer ativa. Ela serve apenas para descobrir o IP atribuído por DHCP.

Lista de objetos treinados

[AQUI](#)

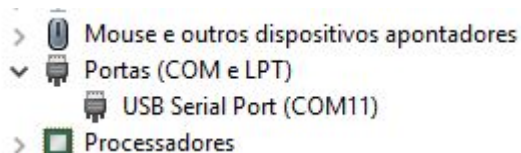


Como funciona o programa?

- 1) Discutiremos isso durante as aulas por meio de análise do código e arquiteturas
- 2) Faremos o debug do código Arduino para entender como funciona
- 3) Faremos o debug no código para entender como a ML funciona
- 4) Discutiremos os conceitos de IA que a disciplina aborda com o código-fonte

À medida que os problemas surgirem, montaremos a lista:

- 1) Testar o cabo de dados conectando um dispositivo no PC e olhar o Gerenciador de Dispositivo

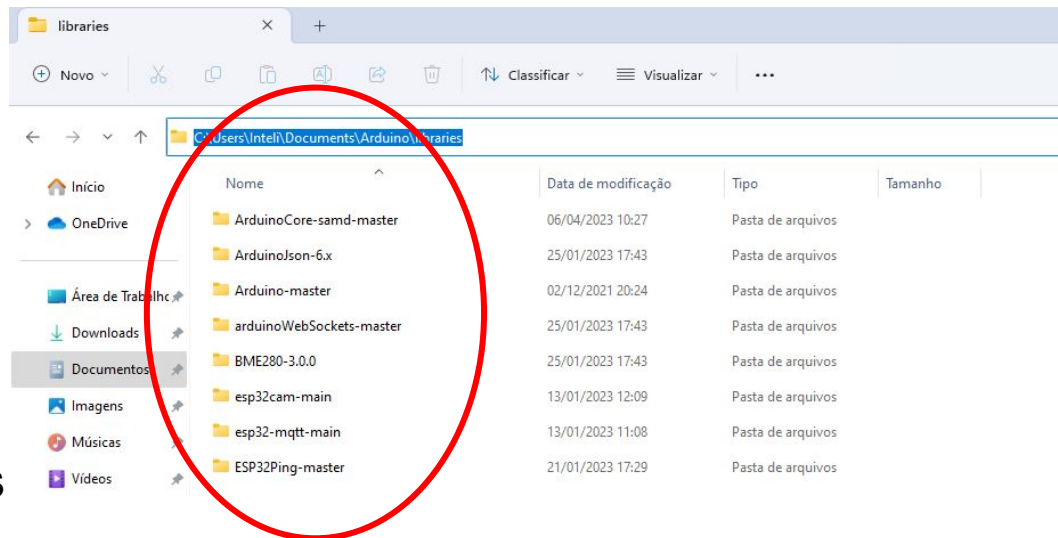


2) Sobre as bibliotecas do ESP32-CAM, vamos removê-las:

a) Vá em DOCUMENTOS/ARDUINO/LIBRARY e apague todas as subpastas que estiverem lá

Veja o exemplo, apague todas as pastas amarelas

b) Depois de deletado, abra e feche o seu Arduino IDE, para ele saber que suas bibliotecas estão zeradas



- 3) Com as bibliotecas removidas, grave um pisca-pisca trocando o LED_BUILINT por 4 (pino do flash). Selecione a placa AI THINKER + porta COM que você já verificou na etapa (1) dessa lista de soluções de problemas;
- a) Garanta que vc executou as ETAPAS (1) a (4) da página 5 desse manual, porque nelas estão a instalação da placa AI Thinker;
 - b) Se a gravação ocorreu com sucesso (100% gravado), mas o flash não pisca, então seu flash queimou mas a placa está OK;

```
Saida
Compressed 18960 bytes to 13073...
Writing at 0x00001000... (100 %)
Wrote 18960 bytes (13073 compressed) at 0x00001000 in 0.6 seconds (effective 269.2 kbit/s)...
Hash of data verified.
Compressed 3072 bytes to 137...
Writing at 0x00000800... (100 %)
Wrote 3072 bytes (137 compressed) at 0x00000800 in 0.1 seconds (effective 472.2 kbit/s)...
Hash of data verified.
Compressed 8192 bytes to 47...
Writing at 0x00000000... (100 %)
Wrote 8192 bytes (47 compressed) at 0x00000000 in 0.1 seconds (effective 667.0 kbit/s)...
Hash of data verified.
Compressed 252624 bytes to 137062...
Writing at 0x00010000... (11 %)
Writing at 0x0001dbb9... (22 %)
Writing at 0x00024c14... (33 %)
Writing at 0x0002a87b... (44 %)
Writing at 0x0002f62e... (55 %)
Writing at 0x00038296... (66 %)
Writing at 0x00040a19... (77 %)
Writing at 0x00046105... (88 %)
Writing at 0x0004baa2... (100 %)
Wrote 252624 bytes (137062 compressed) at 0x00010000 in 3.5 seconds (effective 573.5 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...
```

- c) Se a gravação não ocorreu devido ao print abaixo, vc tem um problema de DRIVER no Windows, isto é, o seu Windows não está usando o driver correto e compatível com a sua placa. Para resolver, vá no slide seguinte

```
Sketch uses 252257 bytes (8%) of program storage space. Maximum is 3145728 bytes.  
Global variables use 22128 bytes (6%) of dynamic memory, leaving 305552 bytes for local variables. Maximum is 327680 bytes.  
esptool.py v4.5.1  
Serial port COM3
```

```
A serial exception error occurred: Cannot configure port, something went wrong. Original message: PermissionError(13, 'Um dispositivo conectado ao sistema não está funcionando.', None, 31)  
Note: This error originates from pySerial. It is likely not a problem with esptool, but with the hardware connection or drivers.  
For troubleshooting steps visit: https://docs.espressif.com/projects/esptool/en/latest/troubleshooting.html  
Failed uploading: uploading error: exit status 1
```

4) O Windows 10 não instalou o drive sozinho. Experimente os driver abaixo nessa ordem. Instalar o driver no modo ADMINISTRADOR

1º CH340 (instale esse primeiro, depois reinicie seu computador)

<https://drive.google.com/open?id=11SoVH3IY9peQrIF8BefhZr1WDBxckUar>

2º CP210x (só instale esse após o 1º não ter funcionado)

<https://drive.google.com/open?id=1XIK8tdi7A9cuXDHROI3P40agcrREV2fq>

O x pode ser qualquer número de 0 a 9

CUIDADO COM O CABO USB. ELE PRECISA TER DADOS E NÃO SOMENTE ENERGIA.
PARA TESTAR O CABO, CONECTE O SEU CELULAR NO PC E AGUARDE O WINDOWS REAGIR.

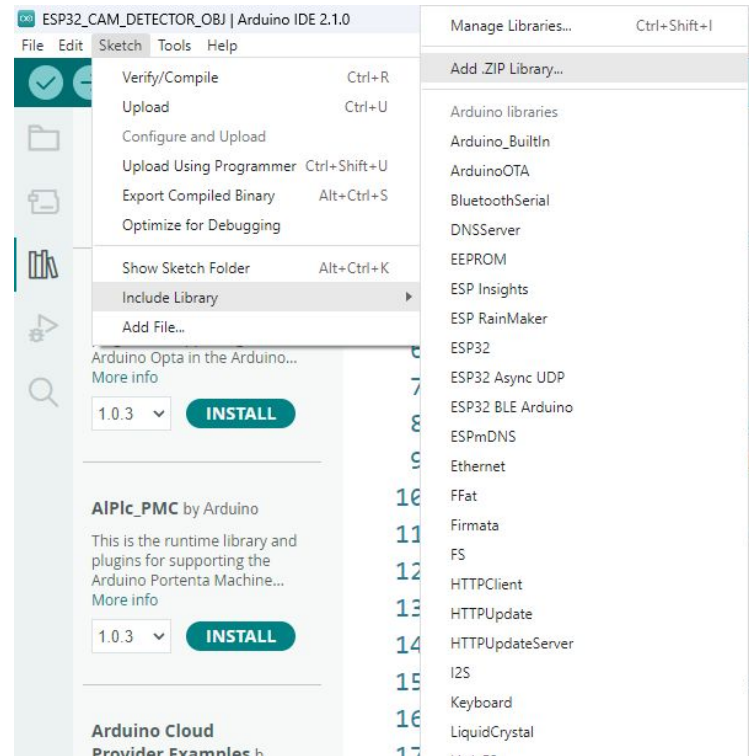
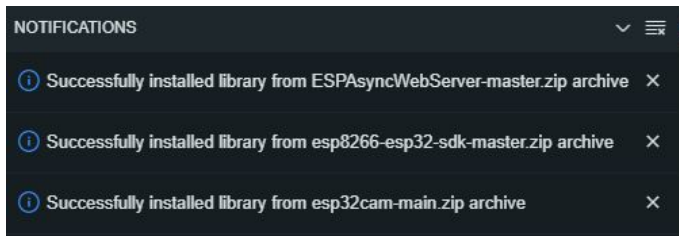
APÓS INSTALAR O DRIVER, REINICIE SEU COMPUTADOR

Solução de Problemas / Parte do ESP32-CAM

5) Após instalado o driver, tente gravar o PISCA-PISCA e agora vai dar certo

6) Vamos retornar a 3 bibliotecas do ESP32-CAM e gravar o programa reconhecedor de objetos.

Passos para reincluir: Sketch → Include Library → Add .ZIP Library... [e selecione o primeiro arquivo *.zip e faça isso 3x, uma para cada biblioteca, separadamente]



- 8) Após instalar as 3 bibliotecas, **feche e abra o Arduino IDE** [importantíssimo]
- 9) Grave o [código-fonte INO](#) do reconhecedor de objetos. Cuidado com as aspas que podem vir com outro formato quando você copiar e colar. As aspas estão nessas linhas

```
const char *WIFI_SSID = "nome da sua rede";  
const char *WIFI_PASS = "senha do seu wifi";
```

- 10) Se após a gravação no ESP32-CAM deu esse erro **CAMERA FAIL** aí no seu Arduino IDE

```
E (513) sccb: SCCB_Write Failed addr:0x30, reg:0xff, data:0x00, ret:-1  
E (514) camera: Camera probe failed with error 0xffffffff(ESP_FAIL)  
CAMERA FAIL
```

Feche e Abra o Arduino IDE como pedido no item (8) aqui e se mesmo assim, persistir, abra o conector da camerezinha, tire o flat cable e recoloque-o e faça o travamento desse conectorzinho preto. Se ainda assim persistir o erro, sua câmera está estragada. A solução é substituir por outra plaquinha.

- 11) Abra o seu MONITOR SERIAL, e deve ficar em branco. Não tem problema! Pressione o botão RESET da placa. Daí o IP vai aparecer. Não se esqueça de ajustar o baud rate para 115200 para não aparecer caracteres estranhos.
- 12) Se após a gravação, a mensagem que aparece é simplesmente **OK CAMERA**, seu roteador é de 5GHz, então, encontre uma alternativa de usar um roteador de 2,4GHz, como o seu celular;

Output Serial Monitor x

Message (Enter to send message to 'AI Thinker ESP32-CAM' on 'COM5')

```

CAPTURE OK 800x600 11092b
e.R.00000200 0026G0j
0000:0000I00I000%Q0 005 0) 000xoL&B0A% 00*0e1A000%00J000nV0p00000*0j0000eC0,k0E00:00(
0+0i00000800,+0:0F00!000 n010L&0000m0S0+0000, 0000i000000K0y00000000F0C0e0.0b080&00000f0
0000000000000000k00s0y 000000G0000E00000000\Er'00000000W0A0i'0'0A0 0000A0000000A0#A00e'':
CAMERA OK
http://192.168.1.8
/cam-lo.jpg
/cam-hi.jpg
/cam-mid.jpg
  
```

Ln 86, Col 2 AI Thinker ESP32-CAM on COM5 5

altere aqui

isso é baud
rate errado

isso é baud
rate certo

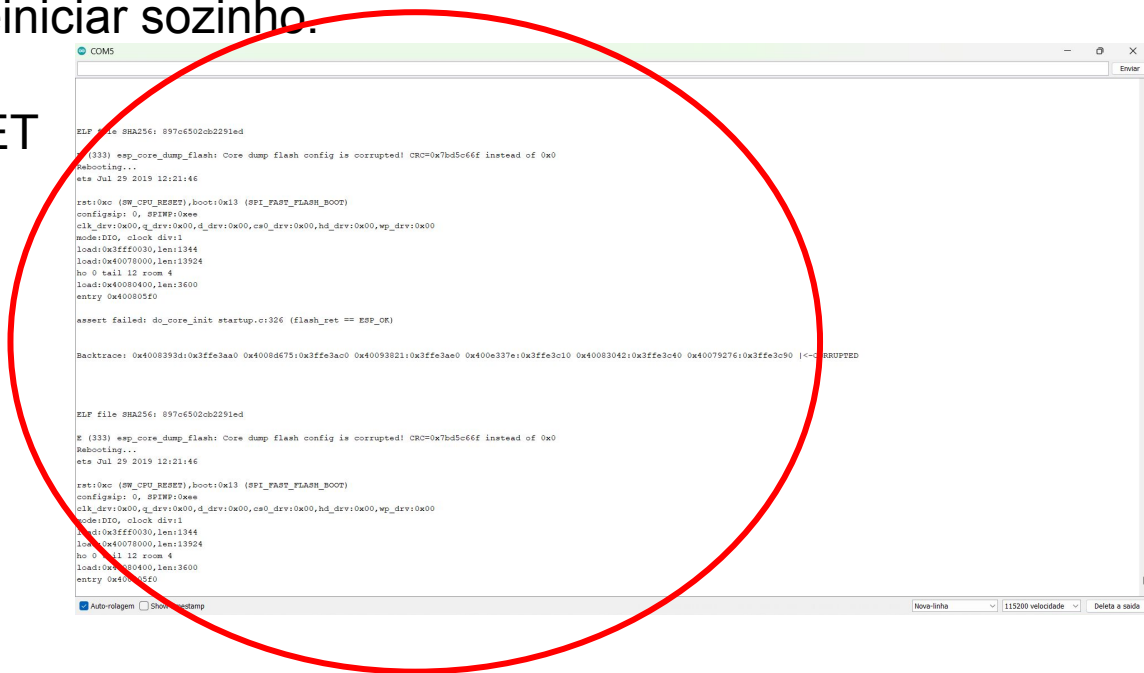
Olha o rodapé e veja se está **ON COMX**

- 13) Se persistir esse erro no MONITOR SERIAL, significa que seu ESP32-CAM não está conseguindo reiniciar sozinho.

Para resolver, aperte o RESET
1x e espera

Ou remova a porta USB
e retorne

Faça algumas vezes até dar
certo



```
COM5

ELF file SHA256: 897c6502cb2291ed

E (333) esp_core_dump_flash: Core dump flash config is corrupted! CRC=0x7bd5c66f instead of 0x0
Rebooting...
ets Jul 29 2019 12:21:46

rst:0xc (SW_CPU_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0030,len:1344
load:0x40078000,len:13924
ho 0 tail 12 room 4
load:0x40080400,len:3600
entry 0x40080500

assert failed: do_core_init startup.c:326 (flash_ret == ESP_OK)

Backtrace: 0x4008393d:0x3ffe3aa0 0x4008d675:0x3ffe3ac0 0x40093821:0x3ffe3ae0 0x400e337e:0x3ffe3c10 0x40083042:0x3ffe3c40 0x40079276:0x3ffe3c90 |<-CORRUPTED

ELF file SHA256: 897c6502cb2291ed

E (333) esp_core_dump_flash: Core dump flash config is corrupted! CRC=0x7bd5c66f instead of 0x0
Rebooting...
ets Jul 29 2019 12:21:46

rst:0xc (SW_CPU_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0030,len:1344
load:0x40078000,len:13924
ho 0 tail 12 room 4
load:0x40080400,len:3600
entry 0x40080500

Auto-rolagem ☐ Show timestamp
```

Se as 2 telas do Python abrem, ficam alguns segundos abertas e pretas, e fecham automaticamente, significa que seu computador e ESP32 não estão na mesma rede WiFi.

Verifique isso antes de bater o desespero!



- 14) Remova todas as bibliotecas do Python. Mas para remover, vai precisar do **pip**

Na tela do CMD (no modo ADMINISTRADOR) garanta que você deu esse comando já dito na pág. 9: *python.exe -m pip install --upgrade pip* ou *-m pip install --upgrade pip*

Começando a remover as bibliotecas:

`pip freeze` → vai listar todas os pacotes

`pip uninstall [nome pacote]==[versão]` → remova os pacotes “sobrando”

Após todas as remoções, reinicie seu PC, e volte na página 9 desse manual e execute as etapas (2) a (5) pulando a instalação do pip.

- 15) Essa lista mostra os pacotes instalados no meu Python

Como remover pacotes diferentes da lista?

- a) primeiro vc dá um pip freeze no CMD
- b) pip uninstall [pacote]==[versão]

Como adicionar os pacotes ao lado?

- c) pip install absl-py==1.4.0 [enter]
- d) pip install astunparse==1.6.3 [enter]
- e) e vai indo

```
C:\Windows\System32>pip freeze
```

```
absl-py==2.2.2  
astunparse==1.6.3  
certifi==2025.1.31  
charset-normalizer==3.4.1  
colorama==0.4.6  
contourpy==1.3.1  
cvlib==0.2.7  
cyclar==0.12.1  
filelock==3.17.0  
flatbuffers==25.2.10  
fonttools==4.55.8  
fsspec==2025.2.0  
gast==0.6.0  
gitdb==4.0.12  
GitPython==3.1.44  
google-pasta==0.2.0  
grpcio==1.71.0  
h5py==3.13.0  
idna==3.10  
imageio==2.37.0  
imutils==0.5.4  
Jinja2==3.1.5  
keras==3.9.2  
kiwisolver==1.4.8  
libclang==18.1.1  
Markdown==3.7  
markdown-it-py==3.0.0  
MarkupSafe==3.0.2
```

```
matplotlib==3.10.0  
mdurl==0.1.2  
ml-dtypes==0.3.2  
mpmath==1.3.0  
namex==0.0.8  
networkx==3.4.2  
numpy==1.26.4  
opencv-python==4.11.0.86  
opt_einsum==3.4.0  
optree==0.15.0  
packaging==24.2  
pandas==2.2.3  
pillow==11.1.0  
progressbar==2.5  
protobuf==4.25.6  
psutil==6.1.1  
py-cpuinfo==9.0.0  
Pygments==2.19.1  
pyparsing==3.2.1  
python-dateutil==2.9.0.post0  
pytz==2025.1  
PyYAML==6.0.2  
requests==2.32.3  
rich==14.0.0  
scipy==1.11.4  
seaborn==0.13.2  
six==1.17.0  
smmap==5.0.2  
sympy==1.13.1
```

```
tensorboard==2.16.2  
tensorboard-data-server==0.7.2  
tensorflow==2.16.2  
tensorflow-intel==2.16.2  
tensorflow-io-gcs-filesystem==0.31.0  
termcolor==3.0.1  
torch==2.6.0  
torchvision==0.21.0  
tqdm==4.67.1  
typing_extensions==4.12.2  
tzdata==2025.1  
ultralytics==8.3.70  
ultralytics-thop==2.0.14  
urllib3==2.3.0  
Werkzeug==3.1.3  
wrap==1.17.2
```

- 16) Se sua tela LIVE TRANSMISSION está OK, mas a tela DETECTION está cinza e não está respondendo, é porque o YOLO (que é o dataset que reconhece objetos) não está corretamente instalado no seu Python. Resolve-se assim:
- a) Confira se sua linha 35 do seu código-fonte Python está assim

```
bbox, label, conf = cv.detect_common_objects(im, confidence=0.6, model='yolov3-tiny')
```

Deixe a linha 35 dessa forma

- b) Vá no seu diretório onde o yolo está instalado e **apague a raiz vermelha**
- C:\Users\SeuNome\.cvlib\object_detection\yolo\yolov3\

Ou seja, vc vai apagar a pasta .cvlib e conseqüentemente, tudo dentro dela vai embora. Quando vc deleta, vc força o Python a reinstalar quando der Play

- 17) Volte no Visual Studio e dá um PLAY no código e verifique que vai começar um download. Espera terminar. Se parar no 99% e der pau no final, não tem problema, vai funcionar mesmo assim.



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

you do not have a GPU set up on your machine.
Iniciado...
2023-05-12 23:49:30.228816: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudart64_110.dll'; dLError: cudart64_110.dll not found
2023-05-12 23:49:30.229433: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
Downloading yolov3.cfg from https://github.com/arunponnusamy/object-detection-opencv/raw/master/yolov3.cfg
Downloading yolov3.weights from https://pjreddie.com/media/files/yolov3.weights
23% |#####|
```


18) Depois, vc pode brincar de alterar a linha 35 fazendo as seguintes combinações:

- a) `bbox, label, conf = cv.detect_common_objects(im, confidence=0.6, model='yolov3')` → modelo completo
- b) `bbox, label, conf = cv.detect_common_objects(im, confidence=0.6, model='yolov3-tiny')` → tiny é mais leve (≈ 2MB de tamanho)
- c) `bbox, label, conf = cv.detect_common_objects(im, confidence=0.6, model='yolov4')` → modelo completo (≈ 251MB de tamanho)
- d) `bbox, label, conf = cv.detect_common_objects(im, confidence=0.6, model='yolov4-tiny')` → tiny é mais leve (≈ 4MB de tamanho)

Mas aguarde o download a cada mudança de `model`

Vc também pode aumentar e diminuir o `confidence` entre 0 e 1 usando a casa decimal. Isso vai aumentar ou diminuir a precisão do reconhecimento, melhorar ou piorar o desempenho. Faça testes no `confidence`

- 19) Se sua tela do Detection abre e fecha sozinho, faça:
- a) Pode ser problema de conexão WiFi, então dá um reset no ESP32-CAM pressionando o botão RST para entrar na rede novamente;
 - b) Confira se o IP na linha 10 do código python bate com o IP do Monitor Serial;
 - c) Confira se seu computador e o ESP32-CAM estão na mesma rede LAN. Redes com proxy geralmente dão problema;
 - d) O ESP32-CAM só entende WiFi de 2,4GHz e não 5GHz, então cuidado ao usar celulares modernos para rotear seu projeto reconhecedor de objetos;
 - e) Apague o diretório C:\Users\SeuNome\.cvlib\object_detection\yolo\yolov3\ conforme já explicado na solução 17.b e dá um play para reinstalar o yolo.
 - f) Confira todas as versões dos pacotes da página 28 deste manual. Tem que bater todas.

- 20) Se sua tela do Detection está detectando você, mas não está detectando objetos tente:
- a) A luz do ambiente está baixa, então clareie mais;
 - b) Use um fundo branco (uma mesa plana branca ou uma folha A4) por baixo de objetos planos e filme esse objeto. Exemplo: garfo, faca, maçã, banana, escova de dente. Conheça a lista completa [AQUI](#)
 - c) Coloque objetos na sua tela de computador e mire a câmera, pois o contraste dos contornos da figura do monitor ajudará;

- 21) E para fechar, você pode comentar a linha 52 com # para não carregar a tela do Live Transmission e seu programar rodar mais fluidamente:

```
#f1= executer.submit(run1)
```

FIM!



E você pode traduzir ou alterar os nomes dos objetos listados no arquivo **yolov3_classes.txt** pra ficar divertido C:\Users\SeuNome\.cvlib\object_detection\yolo\yolov3\yolov3_classes.txt
Por exemplo, no lugar de cat, coloca Milu ou no “dog”, “cuidado! cão bravo”

Yolov3 com alteração de nomes



Alguns objetos reconhecidos

```
classes = ["aeroplane", "bicycle", "bird", "boat", "bottle", "bus", "car", "cat", "chair",  
"cow", "diningtable", "dog", "horse", "motorbike", "person", "pottedplant", "sheep",  
"sofa", "train", "tvmonitor"]
```

Teclado, mouse, frutas, celular, vários outros animais.

No total, são 80 objetos, mas nem todos foram listados aqui, mas estão no slide 2.