

Imagen 1.0(Descarga de Arduino)

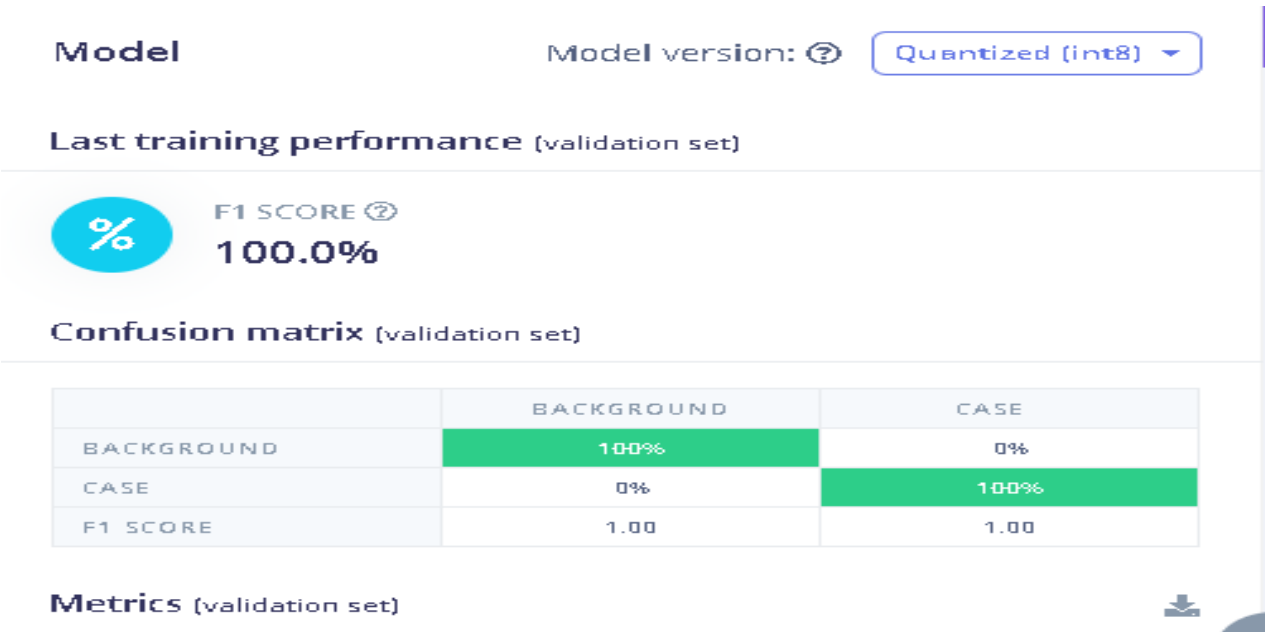


imagen 2.0(Muestra del 100%)

A screenshot of the AI Thinker ESP32-CAM code in the Arduino IDE. The code is written in C++ and is for the AI Thinker ESP32-CAM module. It defines various pins and variables for the camera module. The code is as follows:

```
1 #define EI_CAMERA_RAW_FRAME_BUFFER_COLS 320
2 #define EI_CAMERA_RAW_FRAME_BUFFER_ROWS 240
3 #define EI_CAMERA_FRAME_BYTE_SIZE 3
4
5 /* Private variables ----- */
6 static bool debug_nn = false; // Set this to true to see e.g. features generated from the raw signal
7 static bool is_initialised = false;
8 uint8_t *snapshot_buf; //points to the output of the capture
9
10 static camera_config_t camera_config = {
11     .pin_pwdn = PWDN_GPIO_NUM,
12     .pin_reset = RESET_GPIO_NUM,
13     .pin_xclk = XCLK_GPIO_NUM,
14     .pin_sscb_sda = SIOD_GPIO_NUM,
15     .pin_sscb_scl = SIOC_GPIO_NUM,
16
17     .pin_d7 = Y9_GPIO_NUM,
18     .pin_d6 = Y8_GPIO_NUM,
19     .pin_d5 = Y7_GPIO_NUM,
20     .pin_d4 = Y6_GPIO_NUM,
21     .pin_d3 = Y5_GPIO_NUM,
22     .pin_d2 = Y4_GPIO_NUM,
23     .pin_d1 = Y3_GPIO_NUM,
24     .pin_d0 = Y2_GPIO_NUM,
25     .pin_vsync = VSYNC_GPIO_NUM,
26     .pin_href = HREF_GPIO_NUM,
27     .pin_pclk = PCLK_GPIO_NUM,
28
29     //XCLK 20MHz or 18MHz for OV2640 double FPS (Experimental)
30     .xclk_freq_hz = 20000000,
31     .ledc_timer = LEDC_TIMER_0,
32     .ledc_channel = LEDC_CHANNEL_0,
33
34     .pixel_format = PIXFORMAT_JPEG, //YUV422,GRAYSCALE,RGB565,JPEG
35     .frame_size = FRAMESIZE_QVGA, //QQVGA-LQGA Do not use sizes above QVGA when not JPEG
36
37     .jpeg_quality = 12, //0-63 lower number means higher quality
38     .fb_count = 1, //if more than one, i2s runs in continuous mode. Use only with JPEG
39     .fb_location = CAMERA_FB_IN_PSRAM,
40     .grab_mode = CAMERA_GRAB_WHEN_EMPTY,
41 };
42
43 /* Function definitions ----- */
```

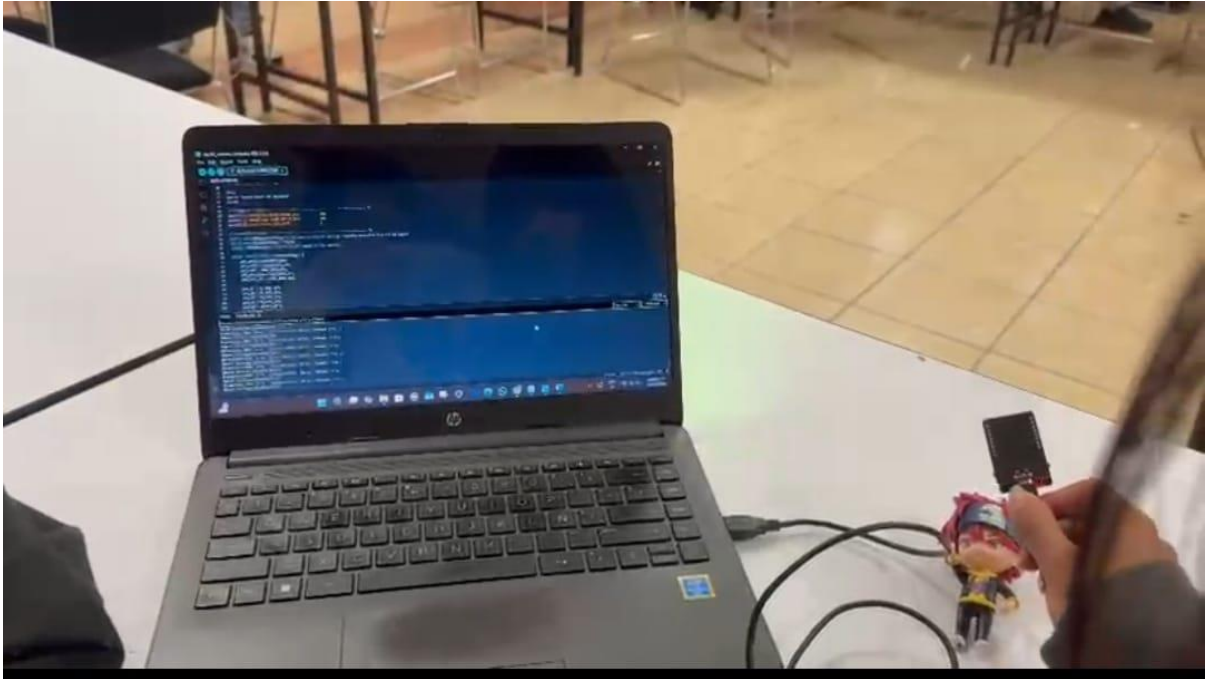


Imagen 4.0 (Detección de objetos)