

# Laboratorio Di Making 5898

## Report Tecnico

Michele Dinelli

September 2025

### Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
<b>2</b>	<b>Componenti Utilizzati</b>	<b>2</b>
<b>3</b>	<b>Collegamento</b>	<b>3</b>
<b>4</b>	<b>Funzionamento</b>	<b>4</b>
4.1	Rilevamento Oggetti . . . . .	4
4.2	Scatto Fotografico . . . . .	4
<b>5</b>	<b>Consumo Energetico</b>	<b>5</b>
<b>6</b>	<b>3D Case</b>	<b>6</b>
6.1	v1 . . . . .	6
6.2	v2 . . . . .	7
<b>7</b>	<b>Sviluppi Futuri</b>	<b>7</b>
<b>8</b>	<b>Valutazione e Critica</b>	<b>7</b>
<b>9</b>	<b>Conclusioni</b>	<b>8</b>

# 1 Introduzione

Questo report descrive lo sviluppo di un prototipo di fototrappola realizzato per il corso Laboratorio Di Making (5898) tenuto durante l'anno accademico 2024/2025 presso l'Università di Bologna.

Una fototrappola è un dispositivo automatizzato, solitamente mimetizzato che rileva il movimento tramite sensori e scatta fotografie quando viene percepito un movimento. Viene utilizzata principalmente per il monitoraggio della fauna selvatica e per la videosorveglianza. È stato scelto come tema del progetto la realizzazione di una fototrappola per curiosità e passione personale, negli ultimi anni ho utilizzato varie fototrappole e ho realizzato, in collaborazione con un fotografo, un archivio online che ne raccogliesse gli scatti. Ho deciso quindi di cimentarmi io stesso nel "costruire" una fototrappola con le risorse a mia disposizione e le conoscenze fornite dal corso.

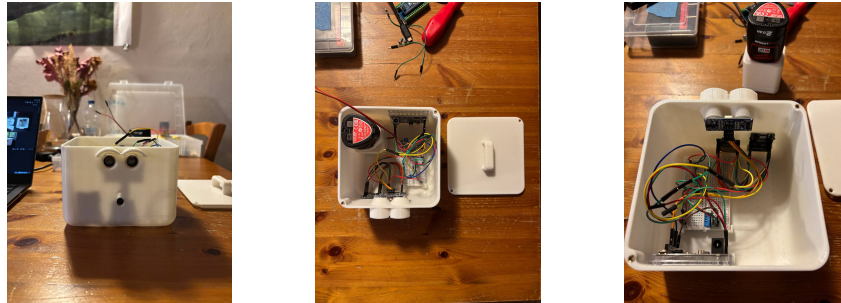


Figura 1: Il prototipo sviluppato.

# 2 Componenti Utilizzati

Il progetto è stato realizzato utilizzando i seguenti componenti:

- Arduino Uno R4 Minima
- ESP32-CAM
  - OV2640 camera (già compresa con la board ESP32-CAM).
  - SanDisk Ultra 32GB microSDHC (acquistata separatamente)
- Sensore a ultrasuoni HC-SR04
- Sensore di temperatura e umidità DHT21
- Cavi di collegamento (jumper cables)
- Resistenze
  - $1 \times 1k\Omega \pm 5\%$

$$- 1 \times 2k\Omega \pm 5\%$$

- Batteria Milwaukee M12 da 2 Ah e 12V

### 3 Collegamento

Il collegamento dei componenti è mostrato in Figura 2 (realizzata con circuit-canvas) . La board Arduino Uno R4 Minima ha collegati i pin 5V e GND sui binari esterni della breadboard. Il sensore HC-SR04 ha i pin VCC e GND collegati sui binari esterni della breadboard, mentre i pin ECHO e TRIG sono collegati rispettivamente ai pin D12 e D11 della board Arduino Uno R4 Minima. Il sensore DHT21 ha collegati i due pin "+" e "-" ai binari esterni della breadboard, mentre il pin OUT è collegato al pin D2 della board Arduino Uno R4 Minima<sup>1</sup>. La board ESP32-CAM ha il pin GPIO12 collegato al pin D7 della board Arduino Uno R4 Minima attraversando due resistenze da  $1k\Omega \pm 5\%$  e  $2k\Omega \pm 5\%$ . Uno dei pin GND della board ESP32-CAM è collegato al pin GND della board Arduino Uno R4 Minima. Il collegamento tra la board ESP32-CAM e la board Arduino Uno R4 Minima è un partitore di tensione che garantisce il corretto funzionamento di ESP32-CAM che lavora a 3.3V mentre Arduino Uno R4 Minima a 5V.

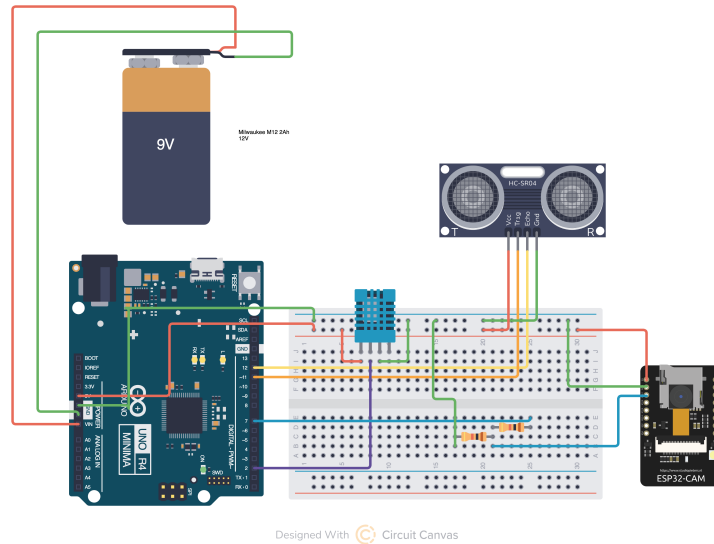


Figura 2: Schema di collegamento dei componenti utilizzati.

<sup>1</sup>Nella rappresentazione dei collegamenti mostrata in Figura 2 il sensore DHT21 presenta 4 pin, questo perché il software per il disegno dello schema non aveva a disposizione la rappresentazione del sensore DHT21 a 3 pin.

Un partitore di tensione è una tipologia di circuito che genera un voltaggio  $V_{\text{out}}$  che è una frazione del voltaggio in input  $V_{\text{in}}$ . Per calcolare  $V_{\text{out}}$  è possibile utilizzare la formula

$$V_{\text{out}} = \frac{R_2}{(R_1 + R_2)} \times V_{\text{in}}$$

Dove  $R_1$  e  $R_2$  sono i valori delle due resistenze. Utilizzando i dati specifici di questo progetto si ottiene

$$V_{\text{out}} = \frac{2k\Omega}{(1k\Omega + 2k\Omega)} \times 5V = \frac{2}{3} \times 5V = 3.\bar{3}V$$

In quanto  $V_{\text{in}} = 5V$  dato dall'input fornito dal pin D7. In questo modo il pin GPIO12 della board ESP32-CAM riceve circa 3,3V in ingresso rispettando il corretto voltaggio richiesto.

## 4 Funzionamento

### 4.1 Rilevamento Oggetti

L'Arduino Uno gestisce il sensore a ultrasuoni HC-SR04 e il sensore DHT21. Il sensore HC-SR04 presenta un trasmettitore e un ricevitore. Funziona inviando onde sonore dal trasmettitore, le quali rimbalzano su un oggetto e ritornano al ricevitore. È possibile determinare quanto un oggetto sia distante misurando il tempo che le onde sonore impiegano per ritornare al sensore. Considerando che la velocità del suono nell'aria è circa pari a 331,4 m/s alla temperatura di 20°C (ossia 0,331 cm/ $\mu s$ ), per calcolare la distanza di un oggetto si può utilizzare la formula:

$$d = \frac{v \times \text{duration}}{2}$$

dove  $v$  è la velocità del suono in cm/ $\mu s$  e *duration* è la durata dell'impulso sonoro in  $\mu s$ . La divisione per due è necessaria perché il segnale compie un tragitto di andata e ritorno. Per stimare meglio la distanza reale, la board Arduino Uno R4 Minima utilizza un sensore DHT21 che misura umidità e temperatura ambientale. Mentre l'umidità ha un'influenza minima sul calcolo della velocità del suono, la temperatura è abbastanza rilevante. Si può infatti correggere la velocità del suono in aria (in m/s) parametrizzandola per temperatura  $t$  e umidità  $h$ :

$$v = 331.4 + (0.6 \times t) + (0.0124 \times h)$$

### 4.2 Scatto Fotografico

Quando un oggetto si trova abbastanza vicino ( $\approx 150$  cm), la board Arduino Uno R4 Minima invia un segnale utilizzando il pin D7 all'ESP32-CAM che lo riceve sul pin GPIO12. Una volta ricevuto il segnale, l'ESP32-CAM avvia il

processo di acquisizione dell'immagine. La foto viene scattata utilizzando una camera OV2640 che cattura immagini a risoluzione SVGA ( $800 \times 600$ ). L'ESP32-CAM genera nomi casuali per le foto e le memorizza su una scheda microSDHC SanDisk Ultra da 32 GB formattata in FAT32.

## 5 Consumo Energetico

Per capire di quanta corrente necessita il progetto e stimarne un limite superiore è possibile consultare il datasheet di ciascun componente e verificare il valore massimo della corrente richiesta da ognuno, quindi sommare i valori. I dati sono raccolti in Tabella 1.

Component	Typ (mA)	Max (mA)
Arduino Uno R4 Minima [1]	33,30	36,98
ESP32-CAM [2]*	180	310
HC-SR04 [3]	15	15
DHT21 [4]	1,5	2,1

Tabella 1: Consumi in mA tipici (Typ) e massimi (Max) dei componenti utilizzati.

Il datasheet di ESP32-CAM riporta un consumo di 180 mA a un voltaggio di 5V a flash spento e 310 mA a 5V con flash acceso. Questo dato rappresenta probabilmente un picco di consumo che si ottiene al momento dello scatto fotografico e viene preso in considerazione come valore tipico 180 mA a 5V in quanto non si utilizza il flash (per non spaventare gli animali). Questo consumo è comunque sovrastimante in quanto non vengono sempre scattate foto e non viene sempre utilizzata la microSD. Inoltre i moduli di Wi-Fi e Bluetooth non vengono utilizzati.

Le board Arduino Uno R4 Minima e ESP32-CAM possono essere entrambe alimentate utilizzando 5V (ESP32-CAM supporta anche alimentazione a 3,3V). Entrambe le board utilizzano un regolatore di potenza e supportano un range di voltaggio in input, Arduino Uno R4 Minima supporta fino a 24V attraverso il connettore a barilotto, ESP32-CAM invece supporta tra i 5V e i 5.25V.

Utilizzando sempre il consumo massimo dei componenti (eccetto per ESP32-CAM) si ottiene:

$$36,98 \text{ mA} + 180 \text{ mA} + 15 \text{ mA} + 2,1 \text{ mA} \approx 234 \text{ mA}$$

Il consumo in Wh si ricava moltiplicando il consumo di corrente per il voltaggio. Considerando un voltaggio pari a 5V si ottiene:

$$5\text{V} \times 0,234 \text{ A} \times 1\text{h} = 1,17 \text{ Wh}$$

Ad esempio, supponendo di utilizzare una batteria da 2000 mAh con tensione a 5V, quindi con capacità di  $5\text{V} \times 2 \text{ Ah} = 10 \text{ Wh}$  la fototrappola avrebbe un'autonomia di

$$\frac{10 \text{ Wh}}{1,17 \text{ Wh}} \approx 8\text{h } 30\text{m}$$

È stato scelto di utilizzare una batteria Milwaukee M12 da 2 Ah a 12V che permette un'autonomia di  $\approx 20\text{h } 30\text{m}$ . La batteria è collegata al pin VIN di Arduino Uno R4 Minima che utilizza un regolatore di voltaggio interno che lo porta a 5V. La board ESP32CAM, così come i sensori usufruiscono dei 5V erogati dal pin 5V di Arduino Uno R4 Minima. A sua volta ESP32CAM ha un regolatore di voltaggio che lo porta a 3,3V.

## 6 3D Case

Per contenere i componenti e organizzare il progetto è stata stampata una scatola con coperchio utilizzando stampa FDM. Per realizzare il modello 3D è stato utilizzato il software Creo, per produrre il file .gcode è stato utilizzato il software per slicing Simplify3d e come stampante la Delta Wasp 2040 Turbo2 con filamento PLA basic bianco da 1.75mm. Ringrazio mio cugino, abilissimo *maker*, che mi ha prestato i software, la stampante e la sua esperienza.

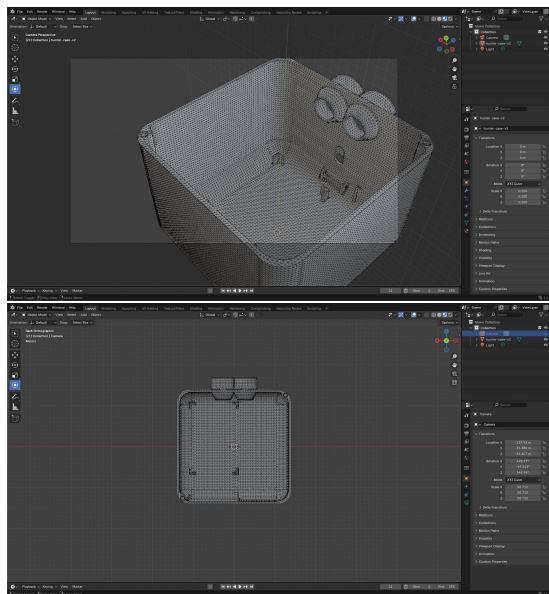


Figura 3: Wireframe della scatola 3D.

### 6.1 v1

I file '.stl' sono disponibili ai seguenti link e possono essere stampati in poche ore.

- [hunter-case.stl](#)
- [hunter-case-cover.stl](#)

I due fori nella parte superiore sono destinati al sensore HC-SR04, mentre il foro nella parte inferiore è per la fotocamera. In origine la fotocamera doveva essere una OV7670 oppure una TTL Serial JPEG Camera, ma per vari motivi "it was not the *case*".

## 6.2 v2

Una seconda versione del case, compatibile con l'ESP32CAM, è disponibile al seguente link [hunter-case-v2.stl](#). È compatibile con il coperchio della versione v1.

## 7 Sviluppi Futuri

In futuro è prevista la sostituzione della board ESP32CAM e sostituirla con una fotocamera seriale e un modulo per la lettura e scrittura su microSD. Il motivo è dovuto dall'eccessivo consumo energetico di ESP32CAM, che nasce per funzioni diverse e più varie rispetto all'utilizzo fatto all'interno di questo progetto.

È possibile migliorare il rilevamento degli animali utilizzando sensori come il TOF10120 che funziona utilizzando laser infrarossi ed è nettamente più preciso del sensore HC-SR04. Viene venduto ovviamente ad un costo maggiore.

Sarebbe anche possibile utilizzare un tool per il design di un PCB shield che renderebbe il progetto compatto e permetterebbe un design 3D più semplice della scatola.

## 8 Valutazione e Critica

La fotocamera utilizzata (OV2640) che come la maggior parte delle fotocamere presenti sul mercato, è dotata di un filtro a infrarossi. Questo è necessario perché aiuta a rimuovere il rumore causato dalla banda dello spettro elettromagnetico degli infrarossi che potrebbe interferire con il valore letto dal sensore che trasforma l'energia elettrica in valori numerici da associare ai pixel. È possibile rimuovere il filtro IR presente sulla OV2640 ma il procedimento rischia di danneggiare la fotocamera se non effettuato con molta attenzione e precisione. La presenza del filtro non permette di ottenere immagini nitide con scarsa luminosità nelle ore notturne, il che limita il progetto in termini di tipologia di animali che possono essere ritratti.

Gli scatti ottenuti dimostrano il funzionamento della fototrappola ma non sono totalmente soddisfacenti per quanto riguarda la qualità delle immagini. Il sensore della camera OV2640 genera direttamente file JPEG compressi e può arrivare a una risoluzione di circa 2 megapixel (UXGA 1600x1200). In teoria

la ESP32CAM dotata di 520KB di SRAM e 4MB di PSRAM <sup>2</sup> dovrebbe poter gestire la massima risoluzione. In realtà la cattura dell'immagine è risultata enormemente problematica ad alte risoluzioni, causando spesso crash-loop. Le motivazioni possono essere svariate e il risultato è stato affidarsi a una risoluzione più bassa (SVGA 800x600) per garantire il corretto funzionamento al prezzo di foto meno nitide.

## 9 Conclusioni



Figura 4: Alcuni scatti.

È stato realizzato un prototipo funzionante di fototrappola con autonomia fino a 20 ore e una scatola con coperchio stampata in 3D per organizzare i sensori e le board utilizzate. Il codice sorgente, la documentazione e i modelli 3D sono open source e rilasciati con licenza GPL-3.0. La repository del progetto è disponibile al link [micheledinelli/cam-ino](https://github.com/micheledinelli/cam-ino).

## Riferimenti

- [1] A. S.r.l., “Arduino uno r4 minima datasheet.” <https://docs.arduino.cc/resources/datasheets/ABX00080-datasheet.pdf>, 2025. Accessed: 2025-09-27.
- [2] Handson Technology, “Esp32-cam module datasheet.” <https://www.handsontec.com/dataspecs/module/ESP32-CAM.pdf>, n.d. Accessed: 2025-09-27.

---

<sup>2</sup>La PSRAM viene usata da ESP32CAM come frame buffer.



- [3] S. Electronics, “Ultrasonic ranging module hc-sr04 datasheet.” <https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>, n.d. Accessed: 2025-09-27.
- [4] H. W. S. . HWSensor, “Hm2301 (dht21) digital-output humidity and temperature sensor — datasheet.” <https://mikroshop.ch/pdf/DHT21.pdf>, n.d. Accessed: 2025-09-27.