

**📸 DIY ESP32-CAM Security Camera with Motion Detection & Telegram Alerts**

This guide will help you build a low-cost smart security camera using **ESP32-CAM** and a **PIR motion sensor**, which sends alerts (with photos) via **Telegram** and also provides **live video streaming** over Wi-Fi.

**🧰 Components Required**

* ESP32-CAM Module (OV2640 camera)
* PIR Motion Sensor (HC-SR501)
* ESP32-CAM USB Programmer
* 2 × 18650 Li-ion Batteries + Battery Holder
* DC-DC Buck Converter (LM2596) – adjustable or fixed 5V
* Breadboard or Custom PCB
* JST Connectors (optional)
* Arduino UNO (for PIR sensor calibration – optional)
* Screws, spacers, double-sided tape
* Multimeter

**🧪 What This Project Does**

* Detects motion using PIR sensor
* Captures a photo on detection
* Sends the photo instantly to your Telegram
* Streams live video from ESP32-CAM over Wi-Fi

**🛠️ Step-by-Step Build Guide**

**✅ 1. Upload Code to ESP32-CAM**

1. Connect ESP32-CAM to USB programmer via header pins.
2. Use Micro-USB to connect it to your PC.
3. Install **Arduino IDE** from Microsoft Store.
4. Add ESP32 Board Support:
   * Go to File → Preferences
   * In **"Additional Board Manager URLs"**, paste:  
     https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\_esp32\_index.json
5. Go to Tools → Board Manager, search **ESP32**, click **Install**
6. Select ESP32 Wrover Module as the board.
7. Open the example: File → Examples → ESP32 → Camera → CameraWebServer
8. Paste your modified code (with Telegram functionality) into this sketch.

**✅ 2. Telegram Setup**

1. In Telegram, search **BotFather**, type /newbot
2. Create a bot → copy the token.
3. Search for **chat ID bot** → start it → copy your Chat ID.
4. Paste your Wi-Fi, Telegram token, and chat ID into the code.

**✅ 3. Final Code Prep**

* Choose a GPIO pin for PIR sensor (example: GPIO 3)
* Set Tools:
  + **Partition Scheme:** Huge App
  + **Board:** ESP32 Wrover Module
  + **COM Port:** (your ESP32’s port)

Click **Upload**.

**📡 4. Live Stream & IP Address**

1. After uploading, open **Serial Monitor**, baud rate: **115200**
2. Press **reset** button on ESP32-CAM
3. Copy the IP address shown — this is your live stream URL.

**🔌 5. Circuit Connections**

* PIR Sensor → ESP32-CAM GPIO (e.g., GPIO 3)
* ESP32-CAM 5V & GND → from **Buck Converter Output**
* Buck Converter Input → from **7.4V (2 × 18650 batteries)**
* Use multimeter to set exact **5V output** on buck converter
* Use double-sided tape, PCB, spacers to neatly mount components

⚠️ Don’t power ESP32-CAM directly from 7.4V — it will burn! Always step it down to 5V.

**🧪 6. PIR Sensor Calibration (Optional with Arduino)**

1. Upload basic motion detection sketch to Arduino UNO
2. Connect PIR sensor’s signal pin to Arduino pin 2
3. Calibrate:
   * Left knob: Sensitivity (distance)
   * Right knob: Time delay (turn **counterclockwise** for short time)
4. Test with onboard LED → once working, use it in final ESP32 circuit.

**📲 7. Testing the Setup**

1. Power the device using batteries or 5V supply
2. Move in front of PIR sensor
3. You should receive a photo alert in Telegram
4. Visit IP address → view live stream

**🛠️ Troubleshooting Tips**

**Problem:** Streaming crashes or lags  
**Fix:**

* Set **XCLK** to 8 MHz in stream settings
* Lower camera resolution