

OV7670

Diana Y.

May 2025

1 Conexiones

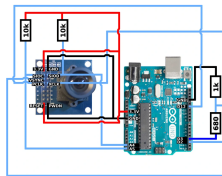


Figure 1: Imagen obtenida de 'How to Use OV7670 Camera with Arduino'

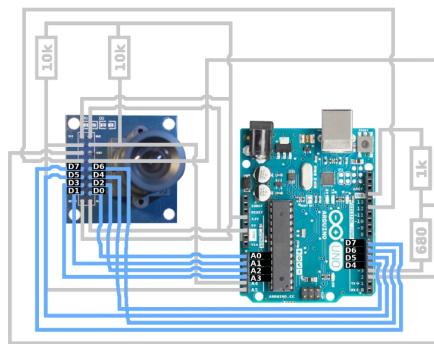


Figure 2: Circuito Estos son los cuatro bits inferiores de un byte de píxel.

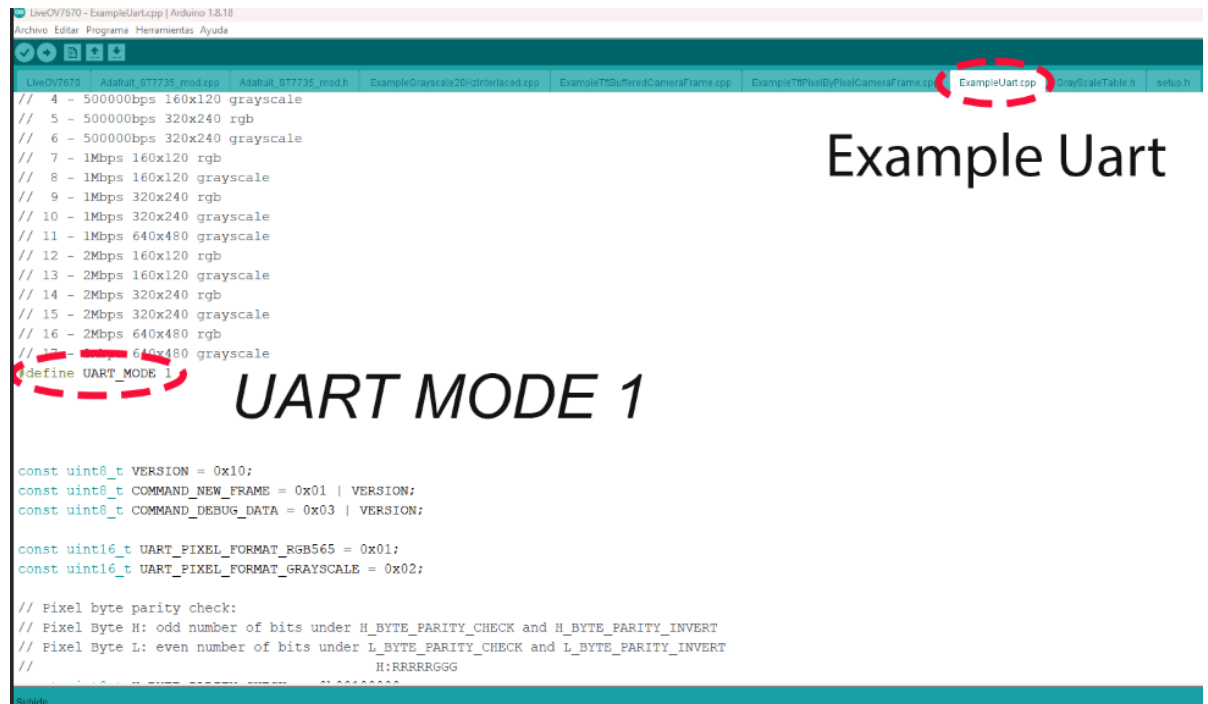


Figure 3: Configuración UART_MODE 1: establece resolución 160x120, formato RGB565 (color) y velocidad de transmisión de 115200 baudios para la comunicación serial con ArduImageCapture y se sube en Setup

Para ejecutar correctamente el clasificador de imágenes en tiempo real, es necesario contar con Python 3.x y las siguientes bibliotecas instaladas. Estas pueden instalarse mediante pip en una terminal con el entorno activo (por ejemplo, usando Anaconda CMD):

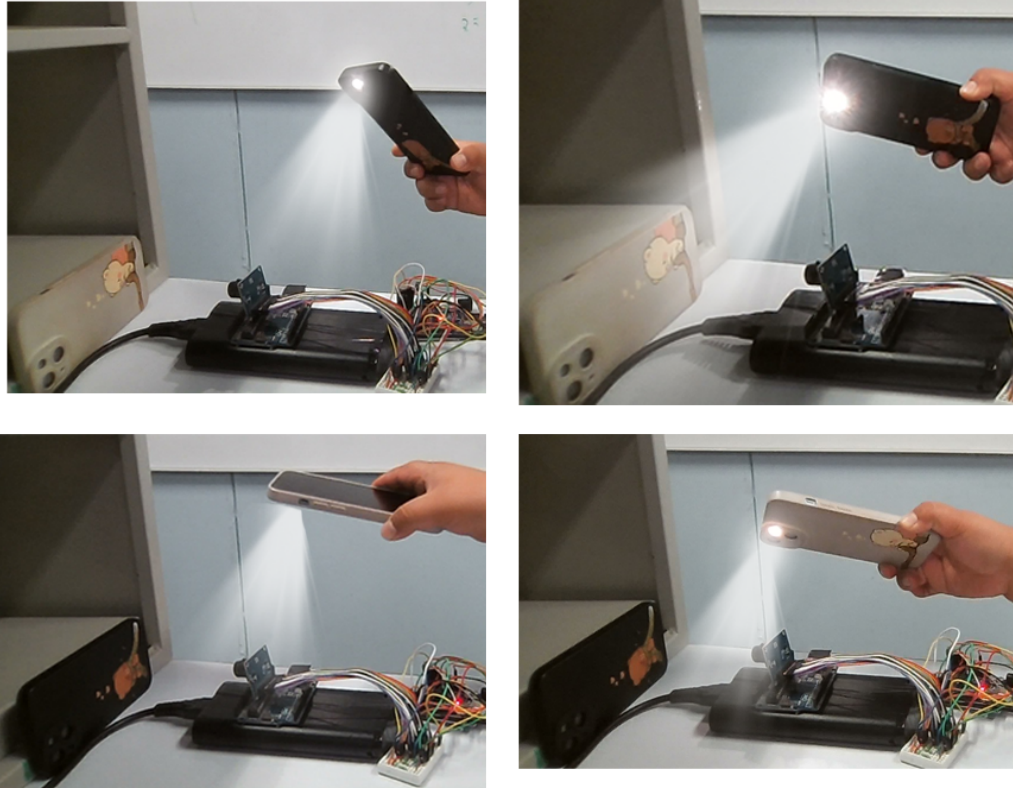


(a) Objeto oscuro



(b) Objeto claro

Figure 4: Comparativa de objetos similares en tamaño y forma, con diferencia de color.



Captura de imágenes utilizando una cámara OV7670 conectada a un microcontrolador. Se aplicaron distintas condiciones de iluminación direccional mediante una lámpara de celular para generar variaciones lumínicas sobre los objetos, manteniendo su posición fija. Estas variaciones enriquecen el conjunto de datos para entrenar un clasificador más robusto. Las imágenes capturadas se almacenaron localmente en `C:/Users/diana/OneDrive/Documentos/arduino-1.8.18/frames`.

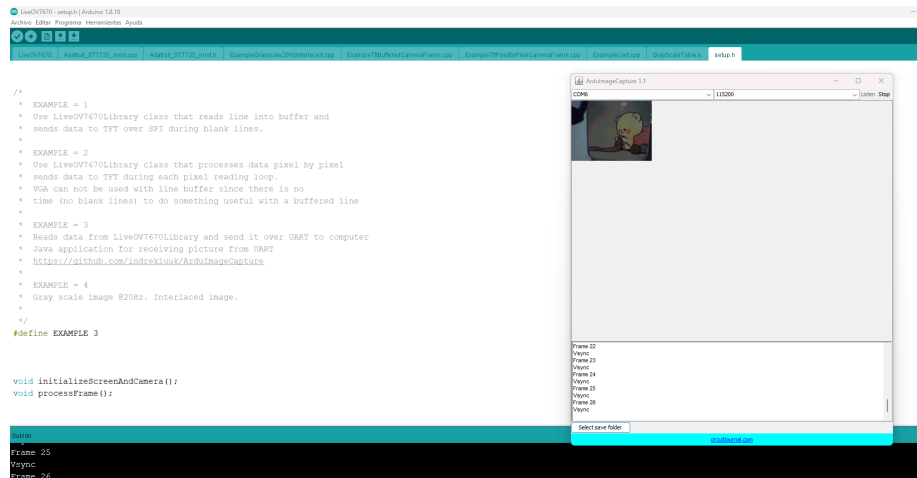


Figure 6: Selecciona la carpeta de guardado con el botón “Select save folder”; las imágenes se guardarán automáticamente en formato .bmp o .png

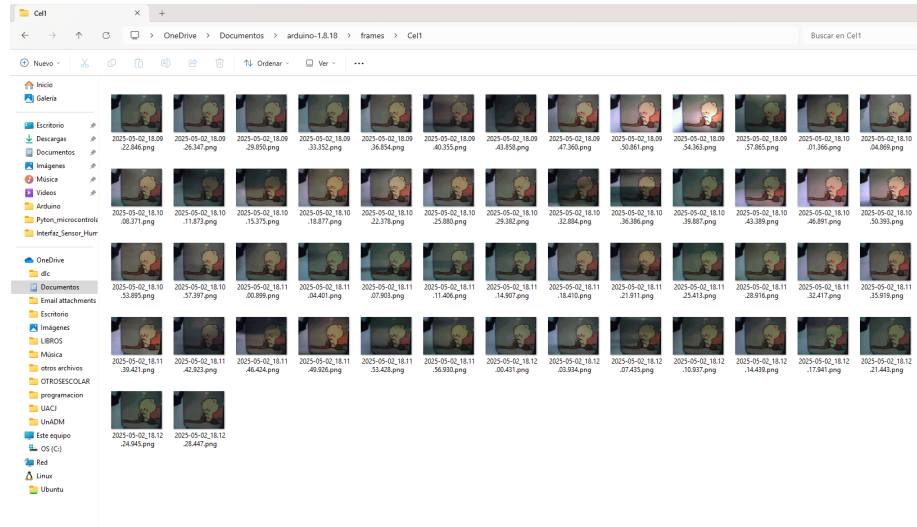


Figure 7: Captura un objeto estático con diferentes condiciones de iluminación. Se utilizó una lámpara de celular moviéndose alrededor del objeto sin desplazar su posición, con el fin de generar un conjunto variado de ejemplos para entrenamiento del clasificador.

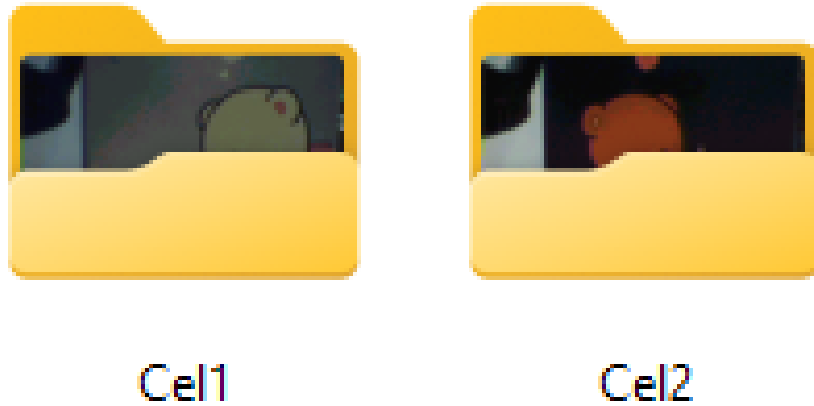


Figure 8: Las carpetas Cel1 y Cel2 contienen conjuntos de imágenes de dos objetos visualmente similares en forma y tamaño, pero con diferente color. Cel1 corresponde al objeto claro y Cel2 al objeto oscuro. Estas imágenes fueron capturadas bajo diferentes condiciones de iluminación para enriquecer el entrenamiento del modelo, localizadas en la ruta de su computadora por ejemplo: `C:/Users/diana/OneDrive/Documentos/arduino-1.8.18/frames`

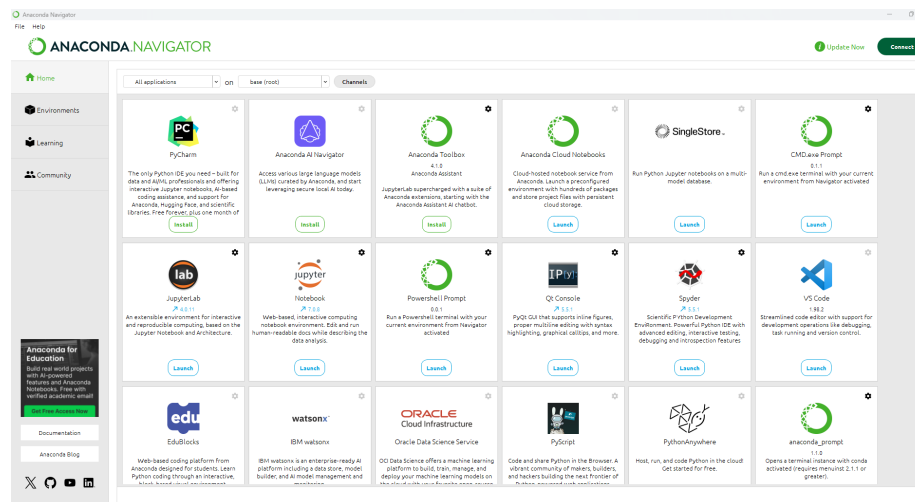


Figure 9: Entorno anaconda

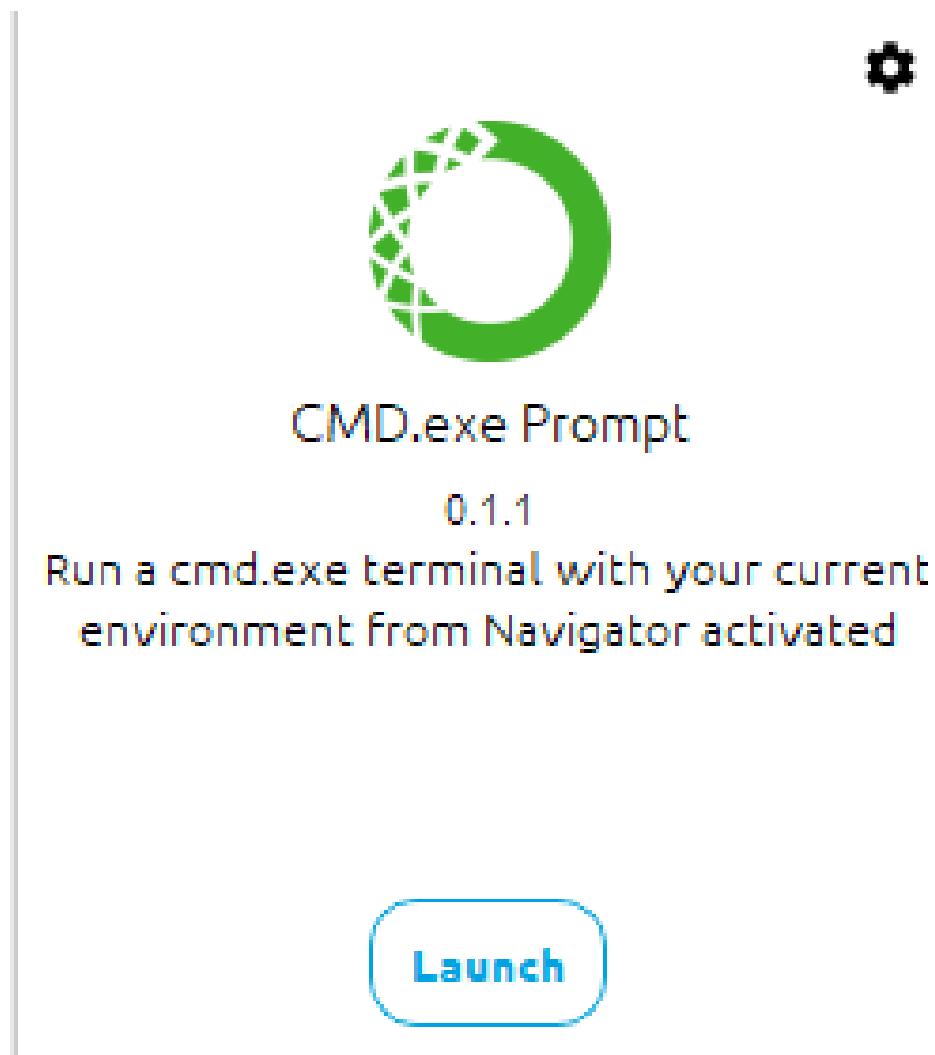


Figure 10: Instala las dependencias necesarias: "pip install opencv-python numpy scikit-learn pillow"



Figure 11: Puedes consultar el siguiente video <https://youtu.be/ErCZbUYwb4E> para instalar las librerías