

Anexo F:

Procesamiento de imágenes

1. Código de visualización de imágenes en tiempo real

Este código utiliza las bibliotecas “maix.display” y “maix.camera” para crear un bucle infinito el cual captura imágenes de la cámara y las visualiza en tiempo real. La instrucción “camera.capture()” captura un fotograma de la cámara y “display.show()” presenta ese fotograma en el dispositivo de visualización. A su vez, el bucle “while True” permite que este proceso de captura y visualización de imagen se repita continuamente, lo que resulta en una transmisión en vivo de la entrada de la cámara en el dispositivo de visualización.

Figura 1.

Código para visualizar imágenes en tiempo real.



Nota: La figura muestra el código de visualizar imágenes en tiempo real luego de ser ejecutado.

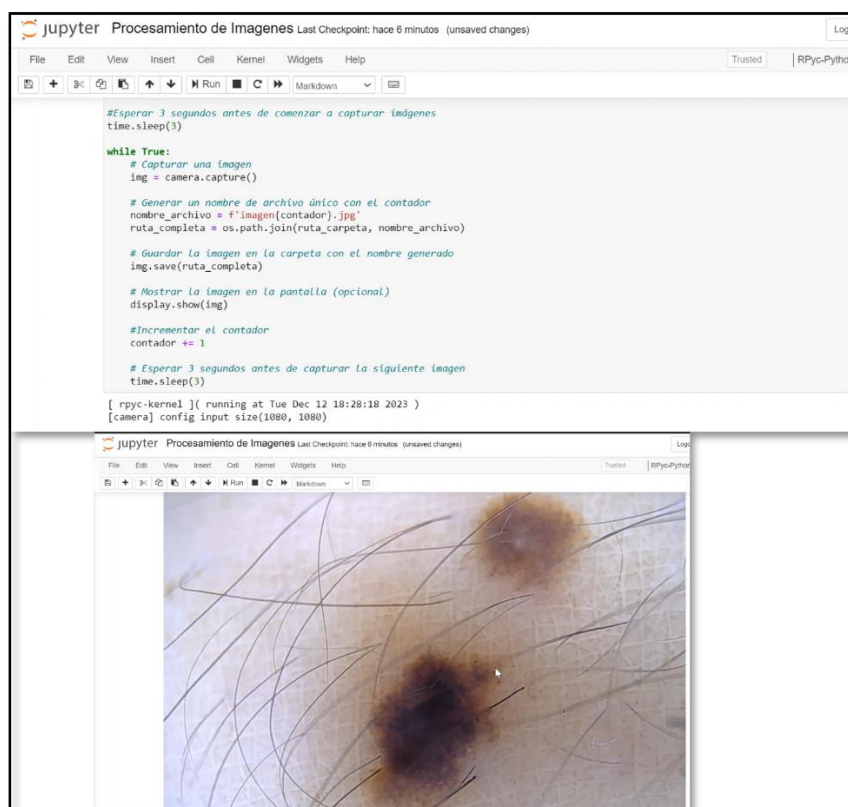
Creado por los autores.

2. Código de captura de imágenes

Este código utiliza las bibliotecas “maix.camera” y “maix.display” para capturar imágenes de una cámara con un tamaño de 1080x1080 en un bucle infinito. Estas imágenes se guardan en una carpeta específica denominada “lizet” en el directorio raíz del sistema. La captura de imágenes se realiza cada 3 segundos, y cada imagen se guarda con un nombre único basado en un contador que se incrementa con cada captura. Además de guardar las imágenes, se muestra cada imagen capturada en la pantalla del dispositivo.

Figura 2.

Imagen capturada de tamaño 1080x1080.



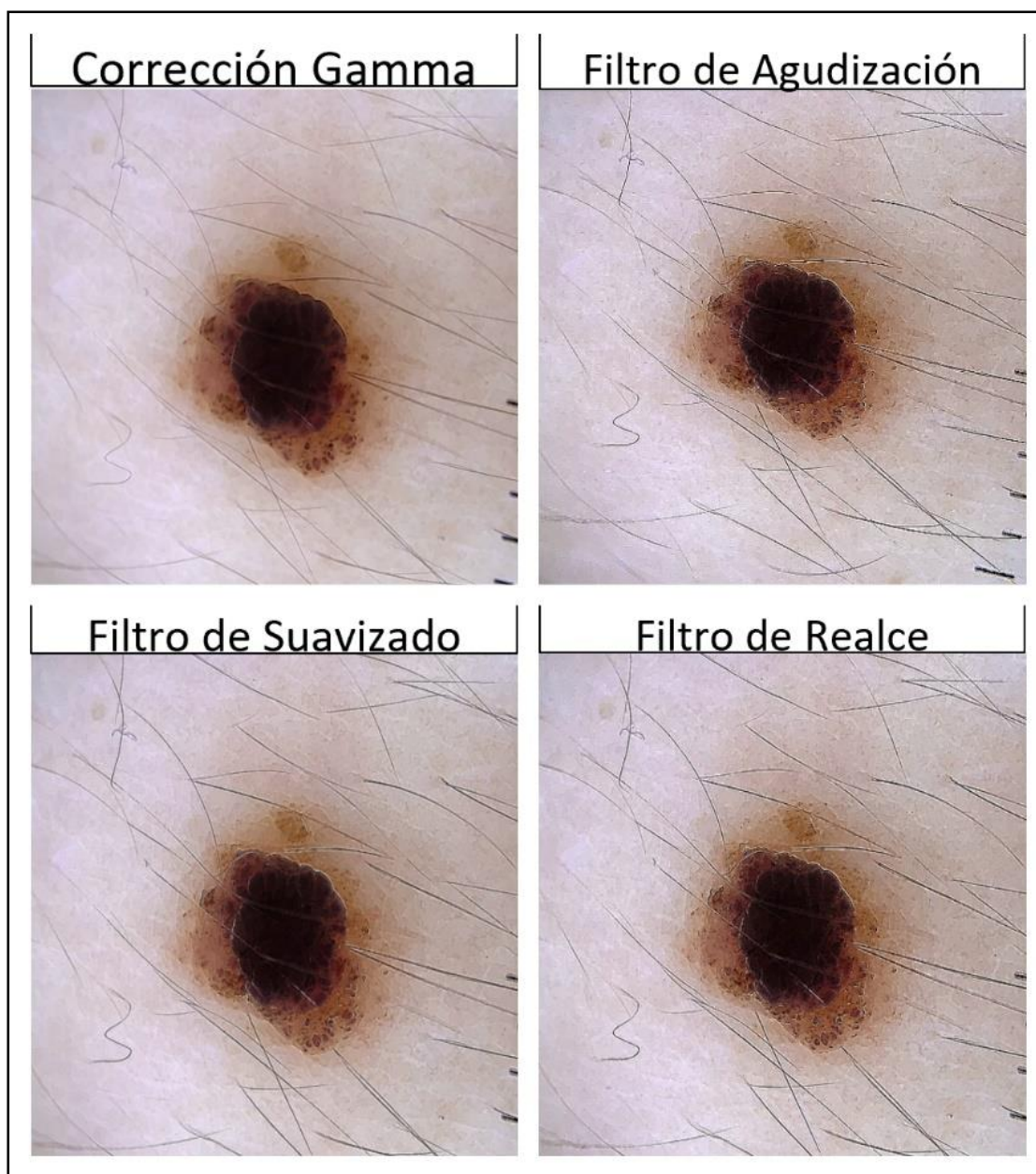
Nota: La figura muestra el código de captura de imágenes luego de ser ejecutado. Creado por los autores.

3. Código de procesamiento de imágenes

Este código utiliza la biblioteca “OpenCV” para cargar una imagen desde una ruta específica. La imagen es procesada en varias etapas, cada una con un propósito diferente. En primer lugar, se realiza una corrección gamma en los canales de color (rojo, verde y azul) de la imagen original, ajustando la intensidad de los píxeles para mejorar la apariencia visual. Luego, se aplica un filtro de agudización para resaltar los bordes y detalles en la imagen corregida. Posteriormente, se aplica un filtro de suavizado para reducir el ruido y suavizar la imagen agudizada. Finalmente, se utiliza un filtro de realce de características para resaltar aspectos específicos de la imagen suavizada. Cada etapa del procesamiento genera una nueva imagen que se guarda en una carpeta ya creada con anterioridad en el módulo. La imagen procesada se guarda con un nombre de archivo único y se imprime un mensaje de confirmación para indicar la finalización de cada etapa del procesamiento.

Figura 2.

Imágenes de respuestas después del procesamiento.



Nota: La figura muestra las imágenes resultantes luego de ejecutar el código de procesamiento de imágenes. Creado por los autores.