

Proyecto 2

1 Descripción

Este proyecto busca integrar diversas técnicas de procesamiento y análisis de imágenes digitales revisadas en el curso en una aplicación básica de rastreo de un objeto bidimensional.

Para ello será necesario utilizar resultados del proyecto 1 para calibrar la cámara utilizada.

2 Objetivo

Al finalizar el proyecto el estudiante habrá implementado un sistema de rastreo de un objeto bidimensional utilizando un proceso de detección de puntos de interés y extracción de descriptores para realizar apareamiento de imágenes de referencia con un *stream* de video.

3 Metodología

El proyecto incluye las siguientes tareas.

1. Investigue e implemente cómo detectar puntos de interés con la OpenCV, así como los descriptores correspondientes. Revise las clases `cv::KeyPoint`, `cv::FeatureDetector` y `cv::SIFT`.
2. Investigue cómo realizar el emparejamiento de descriptores con la OpenCV. Revise las clases `cv::DescriptorMatcher` y `cv::FlannBasedMatcher`.
3. Usted deberá realizar un archivo de configuración que sea leído cuando la aplicación inicie, con opciones que permitan ajustar los parámetros de detección de puntos de interés y los descriptores. Si el archivo de configuración no existe, la aplicación debe generar uno con parámetros por defecto funcionales.
4. Usted deberá indicar a su aplicación, vía la línea de comandos o por medio del archivo de configuración, el nombre de una imagen (y opcionalmente una máscara) que será rastreada durante el resto de la aplicación. La máscara que pase el usuario debe tener el mismo tamaño que la imagen, y de no ser así la aplicación debe detenerse con un error que indique la diferencia de tamaños. Dicha máscara deberá contener el valor “negro” para denotar las partes de la imagen de referencia a ignorar y en “blanco” las partes que interesa rastrear.
5. La aplicación toma la imagen de referencia y le extrae los puntos de interés junto a sus descriptores. Se almacena en las estructuras de datos utilizadas por el emparejador de descriptores únicamente los puntos “activos”, es decir, aquellos que en la máscara no están marcados con negro.

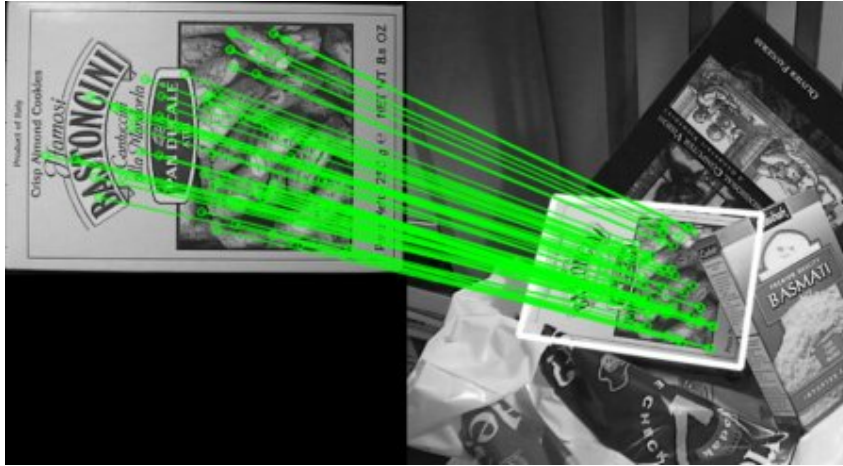


Figura 1: Ejemplo de posible visualización.

6. En la segunda fase de operación la aplicación entra en un ciclo de captura y procesamiento de imágenes. Para cada imagen capturada con el webcam se deben buscar puntos de interés y descriptores, y con ellos hacer el emparejamiento con la imagen de referencia. Debe visualizar dicha correspondencia de algún modo que considere conveniente (ver fig. 1). Para ello hay suficientes videos en YouTube con información al respecto y código disponible en Internet que puede ser tomado como base.
7. Utilice la función `cv::findHomography` para encontrar la homografía (transformación lineal) que mapea los puntos de interés en la imagen de referencia hacia la imagen recién capturada. Use dicha homografía junto a `cv::perspectiveTransform` (u otra función que considere conveniente) para pintar un marco alrededor del objeto rastreado en la imagen que se visualizará para comprobar que los objetos están siendo correctamente rastreados. Observe que usted deberá ignorar correspondencias cuyos descriptores sean suficientemente disímiles.

4 Requisitos

El programa deberá contar con alguna estrategia de captura de la imagen de referencia. Por ejemplo, el programa podría mostrar imágenes en vivo hasta que el usuario presione una tecla, el botón de ratón, etc. en cuyo momento se capturaría la imagen de referencia para continuar entonces con su rastreo.

Puntos extra: utilice la calibración de la cámara del proyecto 1 y los resultados obtenidos en este proyecto 2 para, junto a `cv::solvePnP`, estimar la transformación entre el plano de la referencia y la imagen actual, de modo que se pueda volver a sobreponer al rastreo el mismo sistema coordenado usado en el proyecto 1.

5 Entregables

1. Un reporte corto describiendo el diseño y funcionamiento del proyecto. Procure realizar el reporte científico formal, como se describe [aquí](#).

2. Una demostración del sistema al grupo y al profesor.
3. El código en el formato descrito en el programa del curso.