

## 6. Estimación del flujo óptico.

### Abstracto

La estimación del flujo óptico es un componente importante para aplicaciones donde la información visual cambia dinámicamente con el tiempo. La navegación por robots móviles, el monitoreo del tráfico y el reconocimiento de acciones humanas son tres ejemplos, entre otros. muchos otros, que requieren o pueden beneficiarse de la información del flujo óptico. En esta práctica de laboratorio, implementaremos una sencilla, pero método fundamental para el cálculo del flujo óptico, el algoritmo clásico de Lucas-Kanade.

### Palabras clave

Flujo óptico • Algoritmo de Lucas-Kanade

### Contenido

1 Algoritmo de Lucas-Kanade	1
Cálculo de los gradientes	1
1.2 Calcular $A^T A$ y $A^T b$	1
1.3 Resolviendo el flujo	1
2 Movimiento sintético	1
3 Secuencias reales	2
4 Actividades adicionales y opcionales	2

### 1. Algoritmo de Lucas-Kanade

Nuestra primera tarea es completar la función.

`u,v = flujo_optico(I1, I2, tamaño_ventana, tau)`

para calcular el flujo óptico ( $u$ ,  $v$ ) entre el par de imágenes ( $I1$ ,  $I2$ ) con el algoritmo de Lucas-Kanade (LK), utilizando ventanas cuadradas locales de longitud lateral  $w$  (tamaño\_ventana) y  $\tau$  (tau) como valor mínimo de los valores propios de la matriz  $A^T A$ . vamos proceder paso a paso.

#### Calcular los gradientes

Primero suavizamos las imágenes de entrada con un filtro gaussiano y entonces tenemos que calcular los gradientes espacio-temporales,  $I_x$ ,  $I_y$  y  $I_t$  en las imágenes suavizadas. Para ello definimos los tres núcleos correspondientes para las direcciones  $x$ ,  $y$  y  $t$ ,

y luego aplicar convoluciones 2D. El núcleo y la convolución para  $x$  está dado. **1 Ahora, escribe el código para definir los dos. otros núcleos y aplicar sus respectivas convoluciones.** Para el gradiente temporal, siga esto:  $I_t = (I2 - I1) \cdot M_t$  donde  $M_t$  es la máscara  $4 \times 4$  para un filtro medio. Notación recordatorio: usamos  $\circ$  para denotar la operación de convolución (no una producto matricial o un producto por elementos), y  $I_h$  representa la derivada de la imagen con respecto al eje  $h \in \{x, y, t\}$ .

#### Calcular $A^T A$ y $A^T b$

Recuerde que en el método LK, para cada ventana local de  $n$  píxeles, tenemos que  $Au = -b$ , donde  $A$  es la matriz  $n \times 2$  de

los  $n$  gradientes espaciales, y  $b$  es el vector  $n \times 1$  de temporal derivados. Para conocer el flujo óptico  $u$  para esa ventana, tenga que  $u = (A^T A)^{-1} A^T b$ .

En el código proporcionado, se le proporcionan las matrices  $I_x$ ,  $I_y$  y  $I_t$  que contenga las derivadas correspondientes solo para el ventana local alrededor de un píxel particular en  $(i, j)$ . A partir de estos, **2 solo tienes que calcular las variables**  $A^T A$  (para  $A^T A$ ) y  $A^T b$  (para  $A^T b$ ).

#### Resolviendo el flujo

Finalmente, **3 podemos usar** `linalg.lstsq(C,d)` para calcular el solución de mínimos cuadrados  $x$  a  $Cx = d$ . Sólo presta atención a lo que La matriz  $C$ , el vector  $d$  y  $x$  son en nuestro caso.

Recuerde que sólo podemos calcular el flujo óptico en aquellas ubicaciones de imagen donde  $A^T A$  está bien acondicionado. Entonces, Antes de resolverlo debemos verificar esto. **4 nosotros puede verificar** que el valor propio más pequeño para  $A^T A$  es mayor que  $\tau$ , o que  $A^T A$  tiene rango 2. Para esto, podemos confiar en `np.linalg.eigvals()` y `np.linalg.matrix_rank()`, respectivamente.

### 2. Movimiento sintético

Ahora tenemos que probar cómo funciona el flujo óptico(). Desde el algoritmo básico de LK asume pequeños movimientos, es bueno que podemos generar los movimientos que queremos, bajo control condiciones. Es más difícil encontrar secuencias reales que tengan la movimiento que nos interesa comprobar.

Como puede ver en el código, usamos `SimilarityTransform()` para definir la transformación con las cantidades deseadas de traslación, escala y rotación, que luego se aplica a la transformación real. imagen con `deformación()`.

Para visualizar el flujo calculado, se le proporciona el función `display_optic_flow()`. Buenas visualizaciones de salidas de algoritmos de flujo óptico (y otros) siempre es de importancia crítica durante el desarrollo y análisis de estos algoritmos.

A continuación se puede utilizar el umbral  $\tau = 0.01$ . **5 Experimente con diferentes cantidades de traducción** y encuentre hasta qué grados le va bien a LK. **6 Respecto a**

lo que muestra **la función** `plot_optic_flow()`, asegúrese de comprender, además de las flechas, qué representan los colores del fondo. Compruebe que los resultados se vean cualitativamente buenos (cuando la traducción sea lo suficientemente pequeña). **7 Complemente esta inspección visual** del mapa de flujo con el histograma de la magnitud del flujo óptico. **8 Varíe también el tamaño de la ventana  $w$**  (desde la más pequeña  $w = 3$  hasta unas pocas decenas) y observe el impacto en el flujo estimado (tanto en el mapa como en el histograma de magnitud). ¿Tiene  $w$  influencia en la escasez/densidad o suavidad del mapa? Intente encontrar una explicación.

**9 Repita la experimentación anterior** con el factor de escala (tanto “acercar” como “alejarse”) y luego con las rotaciones.

Observe que, a diferencia de las traslaciones, la magnitud del flujo en diferentes posiciones de la imagen debería ser diferente. Asegúrese de que el resultado tenga sentido, cualitativamente.

### 3. Secuencias reales

Ahora podemos estudiar cómo se comporta el flujo óptico en algunas secuencias de imágenes de escenas reales. Aunque el flujo óptico real difícilmente está disponible en estos casos y, por lo tanto, el error de estimación no se puede calcular fácilmente, de todos modos podemos evaluar el resultado cualitativamente en estos escenarios más desafiantes (e interesantes), donde uno o más objetos pueden moverse de forma independiente, en contraposición al movimiento global de los ejemplos sintéticos anteriores.

**10 Experimente con algunos de los fotogramas consecutivos** de la secuencia del taxi de Hamburgo, que es un ejemplo de las secuencias clásicas en las que solían probarse los métodos de flujo óptico. Aunque el movimiento local en esta secuencia puede considerarse “grande” y, por lo tanto, el método básico LK podría no funcionar muy bien, encuentre para qué tamaños de ventana podemos obtener resultados razonables. ¿Qué beneficio potencial podemos obtener al aumentar  $\tau$  uno o dos órdenes de magnitud?

### 4. Actividades adicionales y opcionales

Recuerde que todos los ejercicios son opcionales y aquellos marcados explícitamente como “opcionales” lo son especialmente. Te proponemos muchos ejercicios opcionales porque no queremos que te quedes con ganas de más actividades. Por supuesto, esto no significa que queramos que los pruebes todos, y mucho menos que los completes todos con éxito. Por lo tanto, sea prudente y selectivo en los ejercicios que desea probar y cuánto tiempo/esfuerzo está dispuesto a invertir, dependiendo de sus antecedentes, su tiempo disponible y sus objetivos. . .

**11 (Opcional) Calcule la pirámide gaussiana de las imágenes de entrada** (puede usar **la pirámide gaussiana**). Experimente un poco con la estimación del flujo en diferentes niveles de la pirámide: como sabe, el flujo óptico se vuelve más pequeño en versiones reducidas de las imágenes, que es la observación básica para la versión de grueso a fino del LK. algoritmo.

**12 (Opcional) Intente segmentar aproximadamente los objetos en secuencias reales** utilizando información de flujo óptico. La idea es utilizar un algoritmo de segmentación simple que conozca (tal vez Otsu), pero en lugar de aplicarlo a valores de nivel de grises, podemos usar flujo óptico. Después de la segmentación, podemos encontrar las regiones (componentes conectados). Por ejemplo, ¿podemos aislar los diferentes vehículos en la secuencia del taxi?

**13 (Opcional) Escriba y pruebe una función para mostrar** información variada sobre el flujo óptico (por ejemplo,  $u$ ,  $v$ , orientación y magnitud). Puede aprovechar esta oportunidad para conocer la flexibilidad de organizar múltiples ejes de Matplotlib en una cuadrícula que ofrece **GridSpec**.

**14 (Opcional) Estudia el `carcuj()`** Función para mostrar la salida del flujo óptico, que se utiliza en nuestra visualización del flujo óptico(). **15**

**(Opcional) Obtenga información sobre una representación de rueda de colores de uso común** que combina la magnitud y la orientación en un único mapa 2D. Puede encontrar ejemplos, por ejemplo, en **este artículo**. Pruebe alguna implementación disponible de este código de colores o impleméntela usted mismo.

Los siguientes ejercicios son más abiertos y posiblemente más difíciles y requieren mucho más tiempo. Algunos incluso están avanzados. Por lo tanto, sólo debe considerarlos si los encuentra, o el tema de esta práctica de laboratorio, motivadores o interesantes, y tiene mucho tiempo libre para dedicarlos. Después de todo, son menos relevantes en el contexto de nuestra introducción.

**16 (Opcional, avanzado) Implemente y experimente con una versión jerárquica de LK de gruesa a fina**. Podemos esperar que se puedan estimar movimientos más grandes.

**17 (Opcional) Pruebe la implementación de Python del LK jerárquico en OpenCV** en algunas secuencias (las utilizadas en esta práctica de laboratorio y/u otras) y compáralas con nuestra implementación básica (de un solo nivel). Quizás también te interese seguir **este tutorial**.

**18 (Opcional) Considere una de las secuencias reales completas** y genere una animación en pantalla (con Matplotlib) o un video atractivo de alguna representación del flujo calculado, ya sea solo o superpuesto con los fotogramas.

**19 (Opcional, avanzado) Estudie algún otro método de flujo óptico bueno y conocido**. Por ejemplo, para quienes tienen inclinaciones matemáticas, en el área previa al aprendizaje profundo, **métodos** eran populares en la visión por computadora. **Brox et al. (ECCV 2004)** propuso un algoritmo variacional para el flujo óptico. Si lo desea, busque también alguna implementación disponible de este (u otro) enfoque y experimente un poco con él. Al menos debería observar que estos mejores métodos superan nuestra implementación simple de LK.

**20 (Opcional, avanzado) Encuentre algún conjunto de datos (pequeño)** con verdad sobre el terreno del flujo óptico disponible. Descubra qué métricas de desempeño se utilizan comúnmente para evaluar el desempeño. Luego, puede ejecutar LK con diferentes métodos  $w$  u otros métodos OF y comparar los resultados con la verdad sobre el terreno utilizando algunas de esas métricas.