

Seguimiento inteligente: modelando movimiento de objetos en cámaras de video

Smart tracking: modeling object motion in video cameras

Ángel J. Sánchez-García¹

Resumen: El seguimiento y modelado de objetos en movimiento es una tarea fundamental dentro del campo de la visión artificial, con aplicaciones en videovigilancia, vehículos autónomos, monitoreo de tráfico y más. Este texto presenta una metodología integral que incluye segmentación, análisis de flujo óptico y modelado mediante el filtro de Kalman. Se inicia con la segmentación de objetos en una escena, utilizando características como color, forma o textura. Posteriormente, se aplica el flujo óptico para detectar movimiento entre *frames* consecutivos, superando desafíos como el problema de la apertura mediante la detección de esquinas. Los vectores de movimiento se agrupan para formar regiones, y se define una cubierta convexa para representar cada objeto. Finalmente, el filtro de Kalman permite estimar posiciones futuras con base en mediciones anteriores, mejorando la precisión del seguimiento. Esta metodología permite adaptar el seguimiento a distintos entornos, anticipar comportamientos y facilitar la toma de decisiones automáticas en tiempo real.

Palabras clave: visión por computadora, seguimiento, movimiento, flujo óptico, filtro de Kalman

Abstract: Tracking and modeling moving objects is a fundamental task within the field of computer vision, with applications in video surveillance, autonomous vehicles, traffic monitoring, and more. This chapter presents a comprehensive methodology that includes segmentation, optical flow analysis, and Kalman filter modeling. It begins with the segmentation of objects in a scene, using features such as color, shape, or texture. Optical flow is then applied to detect motion between consecutive frames, overcoming challenges such as the aperture problem through corner detection. Motion vectors are grouped to form regions, and a convex hull is defined to represent each object. Finally, the Kalman filter allows future positions to be estimated based on previous measurements, improving tracking accuracy. This methodology makes it possible to adapt tracking to different environments, anticipate behaviors, and facilitate automatic decision-making in real time.

Keywords: computer vision, tracking, motion, optical flow, Kalman filter.

¹ Doctor en Inteligencia Artificial. Universidad Veracruzana. angesanchez@uv.mx

Introducción

En la actualidad, no se puede concebir el mundo sin tecnología, pues esta se encuentra ya en todas partes. Las nuevas generaciones crecen con dispositivos inteligentes que permiten gestionar la vida cotidiana; desde recomendaciones para ver una película, asistentes inteligentes para escuchar música y *chat bots* que responden a muchas de las preguntas que se tienen. Avances tecnológicos se han dado gracias al desarrollo y crecimiento del área llamada inteligencia artificial, la cual proporciona muchos elementos para tomar decisiones de manera automática.

En las últimas décadas, el campo de la visión artificial ha experimentado una evolución importante, permitiendo a las computadoras percibir, interpretar y comprender la información visual de maneras que antes se consideraban ciencia ficción. Dentro de las actividades más relevantes en este campo se encuentran el seguimiento y modelado de objetos en movimiento en video. Este tema se encuentra en la intersección de varias disciplinas, como la visión artificial, el procesamiento de imágenes, el aprendizaje automático y la robótica, y desempeña un papel vital en una amplia gama de aplicaciones del mundo real, como la vigilancia, la conducción de vehículos autónomos, la monitorización del tráfico y la realidad aumentada.

En los sistemas de videovigilancia, esta capacidad es fundamental para monitorear espacios públicos, detectar comportamientos anómalos (Liu et al., 2018) (Parthasarathy et al, 2019) y garantizar la seguridad en tiempo real (Sigh et al., 2018)(Garje et al., 2018) (Sarcar & Yousuf, 2019). Este comportamiento anómalo dependerá de cada ambiente y sus características. Por ejemplo, el detectar movimiento de objetos muy rápido en alguna plaza o banco genera un movimiento anómalo que puede percibirse como alguna alarma de robo, mientras que en hospitales pudieran interpretarse los movimientos rápidos como alguna caída de algún infante o de algún adulto mayor. Finalmente, si se está monitoreando los objetos que pasan a través de una carretera, es posible identificar aquellos que se mueven más rápido y, por lo tanto, notificar algún peligro de un potencial accidente vehicular. Como podría intuirse, la velocidad que puede ser tolerable en cada uno de estos tres escenarios es diferente; sin embargo, la tarea es la misma: identificar objetos que se mueven de manera sospechosa.

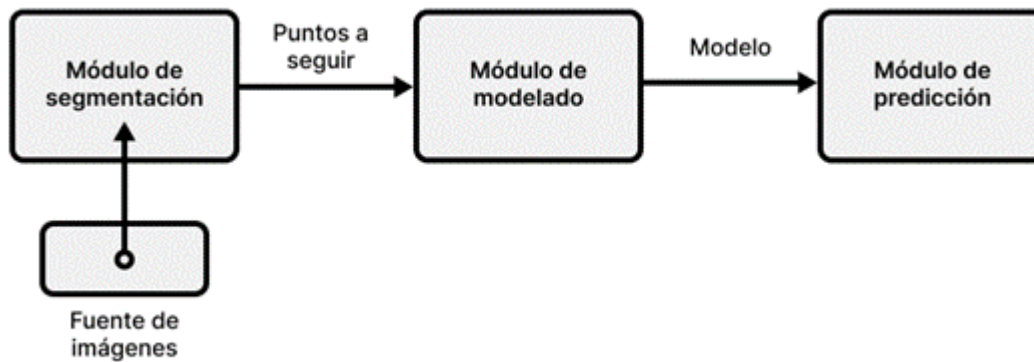
El presente capítulo busca dar un panorama general de la metodología para realizar el proceso de modelado y seguimiento de objetos en movimiento; con el fin de poder aplicarlo a diferentes ambientes, dados los parámetros y circunstancias de cada escena en particular.

Metodología

Para realizar este seguimiento de objetos en movimiento es necesario abordar dos conceptos principales: segmentación y modelado. Estas tareas son indispensables

para tener estimaciones precisas del movimiento de uno o más objetos en una escena. La figura 1 muestra el panorama del proceso dividido en 3 módulos principales: módulo de segmentación, módulo de modelado, y módulo de predicción.

Figura 1 Método para el seguimiento de objetos en movimiento



Módulo de segmentación

Un paso esencial en el análisis de objetos basados en contenido es realizar una separación espacial de un objeto, del contexto de la imagen actividad cuyo proceso se denomina segmentación. Si bien es cierto que la identificación de los límites que delimitan un objeto es sencillo de identificar para el ojo humano, hacerlo de manera automática es una tarea no trivial (Garcia et al., 2008). En el módulo de segmentación se recibe un conjunto de imágenes (*frames*) de una fuente de video que servirá principalmente para identificar los objetos de interés del ambiente. Existen tres formas tradicionales para segmentar e identificar objetos en una sola imagen, a partir del color de la forma o de la textura.

Hay casos en los que se requiere conocer la ubicación de objetos de un color específico, de modo que los píxeles vecinos que comparten un color similar se agrupen (Littmann & Tirret, 1997)(Fuh et al., 2000)(Wu et al., 2000)(Fleye, 2004). Los objetos también pueden identificarse en una imagen con una forma predefinida; por ejemplo, es posible encontrar los píxeles que forman un círculo mediante la transformación de Hough (Ballard, 1981)(Leavers, 1993). Finalmente, las regiones suelen identificarse si comparten la misma textura o una similar, mediante diversos métodos, como transformaciones wavelet (Lu et al., 1997), filtros multicanal (Farrokhnia & Jain, 1991) y recientemente filtros de Gabor (Wang et al., 2008)(Bitam & Ameer, 2013).

Sin embargo, si se requiere identificar objetos en movimiento, es necesario contar con, al menos, dos imágenes. Esto se debe a que se necesita saber si hubo algún cambio

en la intensidad de los píxeles entre las dos imágenes, lo que permite identificar un movimiento.

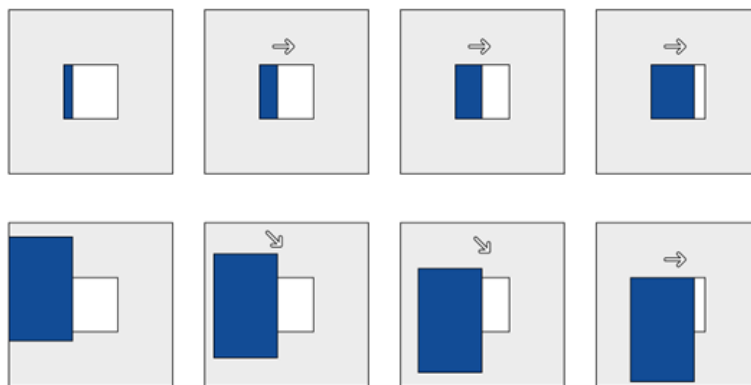
Flujo óptico

Para identificar objetos que se mueven de una escena a otra, se suele utilizar el flujo óptico. Horn y Schunck (1981) definen el flujo óptico como la relación de los cambios espaciotemporales en la intensidad de los píxeles entre dos imágenes; es decir, para encontrar la ubicación de los píxeles cuya intensidad cambió de una imagen a otra. Para calcular el flujo óptico, los métodos habituales pueden clasificarse en métodos densos, como el método de Horn y Shunk (1981), y dispersos, como el método de Lucas-Kanade (1981). Este último se basa únicamente en la información local de cada píxel, derivada de una pequeña ventana del píxel de interés; a diferencia del método de Horn y Shunk, que se basa en la información global de la imagen. A veces puede ser necesario no solo identificar objetos en movimiento, sino que, una vez identificado el objeto, puede ser importante comprender cómo este se mueve; es decir, se pueden obtener la dirección y la magnitud del movimiento.

Puntos de interés

El flujo óptico no se puede identificar en todos los píxeles de la imagen, ya que algunos no proporcionan información para su ubicación en la imagen siguiente (tiempo $t+1$). El flujo óptico normal resulta del problema de apertura, que surge cuando se tiene una abertura o ventana pequeña para medir el movimiento. Cuando se detecta movimiento con una abertura pequeña, a menudo solo se ve un borde, no una esquina. Sin embargo, un borde por sí solo no es suficiente para determinar con exactitud cómo (es decir, en qué dirección) se mueve todo el objeto (Bradski, 2008) como se observa en la figura 2, donde se muestra el problema que la apertura genera a través de una ventana de apertura (fila superior): se ve un borde que se mueve hacia la derecha, pero no es posible detectar la parte descendente del movimiento (fila inferior).

Figura 2 Problema de la apertura. Fuente: (Bradski, 2008)



Para evitar este problema, deben encontrar píxeles que sean muy diferentes de sus vecinos. Harris & Stephens (1988) los denominan esquinas. Las esquinas presentan cambios significativos de intensidad en direcciones ortogonales como característicos. Es posible utilizar las derivadas de segundo orden de las imágenes, tomadas en todos los puntos de la imagen; por ejemplo, la matriz Hessiana alrededor de un punto (p), que se define en dos dimensiones por las intensidades de la imagen.

Ecuación 1. Cálculo de la matriz Hessiana para cada pixel

$$H(p) = \begin{bmatrix} \frac{\partial^2 I}{\partial x^2} & \frac{\partial^2 I}{\partial x \partial y} \\ \frac{\partial^2 I}{\partial y \partial x} & \frac{\partial^2 I}{\partial y^2} \end{bmatrix}$$

Shi y Tomassi en (1994) definen los puntos como “Características útiles para el seguimiento” basándose en la definición de la matriz de Harris y tomando como criterio para determinar si son vértices el cálculo de los autovalores de dicha matriz. Dado que la matriz de autocorrelación es cuadrada y de orden 2, se obtienen dos eigenvalores. Al calcular los eigenvalores λ_1 y λ_2 , se obtiene uno de tres casos posibles.

Si $\lambda_1 \approx 0$ y $\lambda_2 \approx 0$, el píxel no contiene información relevante.

Si $\lambda_1 \approx 0$ y λ_2 tiene un valor positivo grande, se trata de una arista.

Si λ_1 y λ_2 son valores positivos grandes, el píxel es una esquina.

Si el valor propio mínimo supera un umbral α , la esquina tiene un valor fiable para su identificación.

Agrupación de vectores

El seguimiento de objetos implica la identificación y localización consistente de objetos en *frames* consecutivos de una secuencia de video. El objetivo es mantener la identidad de cada objeto a pesar de posibles cambios de apariencia, oclusiones o variaciones de movimiento.

Por esta razón, tras obtener los vectores de flujo óptico, se pueden agrupar por similitud. Para crear cada región que define cada objeto, es necesario agrupar los vectores de flujo óptico según tres criterios de un vector: proximidad, dirección y magnitud.

1. $P(x_1, y_1)$ y $Q(x_2, y_2)$ son los puntos donde comienzan los vectores A y B respectivamente (puntos de interés). A y B se consideran vectores cercanos si los puntos P y Q no difieren tanto en x y y de un umbral β , es decir:

$$\text{abs}(x_2 - x_1) < \beta$$

$$\text{abs}(y_2 - y_1) < \beta$$

2. A y B se consideran vectores con dirección similar si el ángulo entre ambos vectores no excede un umbral γ :

$$\angle(A, B) < \gamma$$

3. A y B se consideran vectores similares en términos de su magnitud, si la diferencia entre sus magnitudes no difiere de un umbral δ , es decir:

$$\text{abs}(\|A\|, \|B\|) < \delta$$

Segmentación de objetos en movimiento

Para cada grupo de vectores, se obtiene su cubierta convexa para abordar el problema de la discontinuidad de las regiones de flujo óptico y segmentar la región (Sánchez et al., 2014). Los puntos utilizados para generar la envoltura convexa de cada región son los puntos de partida de los vectores pertenecientes a la región. Es posible representar la condición para determinar si dos puntos pertenecen a la envoltura convexa del conjunto de puntos S. Dos puntos $P(x_1, y_1)$ y $Q(x_2, y_2)$ pertenecen al conjunto de la envoltura convexa C si y solo si todos los puntos $R(x_3, y_3)$ pertenecientes a S (excepto P y Q), al evaluarse en la ecuación de la recta que pasa por los puntos P y Q, están a un lado de la recta (el signo debe tener un único valor para cada punto R, ya sea positivo o negativo).

Finalmente, después de crear una cubierta convexa en cada frame es necesario encontrar un punto representativo de cada cubierta debido a que las formas pueden cambiar a través del tiempo. A decisión del diseñador puede optarse por obtener el promedio o la mediana de cada convexa como punto representativo a seguir en cada uno de los frames.

Módulo de modelado

Por otro lado, el modelado añade un nivel de abstracción al representar el movimiento, el comportamiento y la estructura de los objetos mediante modelos matemáticos o basados en datos. Juntos, el seguimiento y el modelado proporcionan a las máquinas una comprensión más profunda de su contexto visual, lo que les permite no solo detectar movimiento, sino también anticipar posiciones futuras, interpretar intenciones e interactuar adecuadamente con entornos dinámicos.

Este módulo recibe como entrada todos los puntos obtenidos en el módulo de segmentación para generar un modelo que permita pronosticar el siguiente punto o ubicación para que pueda seguirse a través del tiempo. De hecho los vectores de flujo óptico son muy útiles porque evalúan hacia dónde se ha movido el objeto a través del tiempo y a dónde debería continuar moviéndose si es que este no cambia de dirección de manera brusca.

Filtro de Kalman

El filtro de Kalman es un conjunto de ecuaciones matemáticas, descrito por primera vez en 1960, donde se presenta una solución recursiva al problema del filtrado lineal de datos discretos. Este método ha sido ampliamente investigado y aplicado en diversos campos, ya que proporciona un mecanismo computacional (recursivo) eficiente para estimar el estado de un proceso. El filtro es potente porque implica estimaciones de estados pasados, presentes e incluso futuros. El filtro de Kalman integra estos elementos mediante el uso del conocimiento del sistema y del dispositivo de medición, la descripción estadística de los ruidos del sistema y cualquier información disponible sobre las condiciones iniciales de las variables de interés (Maybeck, 9179).

Para comenzar, se debe recordar que la idea básica del filtro de Kalman es que, bajo un conjunto de supuestos, será posible, dado un historial de mediciones de un sistema, construir un modelo para el estado del sistema que maximice la probabilidad *a posteriori* de las mediciones previas.

Es posible maximizar la probabilidad *a posteriori* sin tener un largo historial de mediciones. En cambio, es posible actualizar iterativamente el modelo de estado de un sistema y mantener únicamente ese modelo para la siguiente iteración (Bradski, 2008). Estas iteraciones se forman principalmente mediante procesos de predicción, medición y actualización del estado.

Sin embargo, se debe precisar que el filtro de Kalman se basa en tres supuestos:

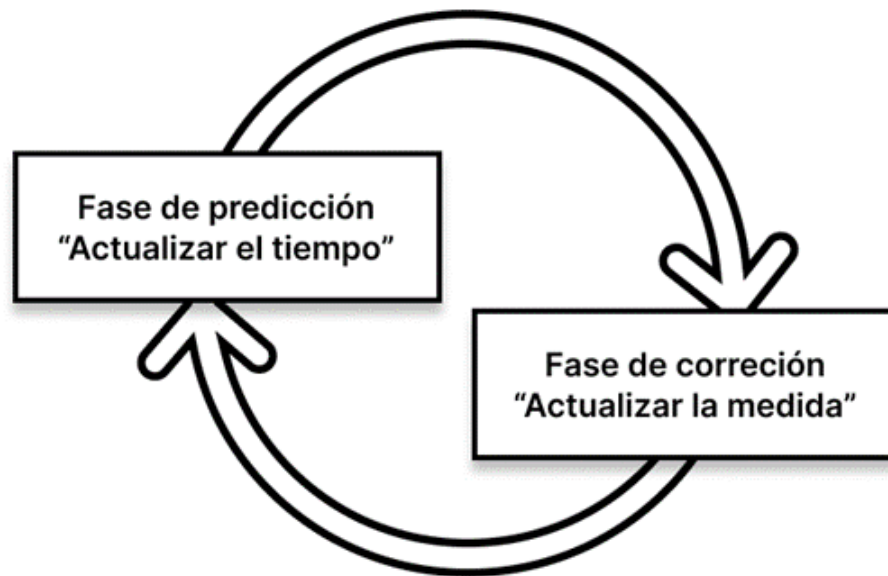
1. La evolución del espacio de estados es lineal.
2. Los errores o ruidos sujetos a las mediciones son “blancos”.
3. Este ruido también es gaussiano.

En otras palabras, el primer supuesto implica que el estado del sistema en el tiempo t puede modelarse como una matriz multiplicada por el estado en el tiempo $t - 1$. Esto es positivo porque los sistemas lineales son más fáciles de manipular y prácticos que los no lineales (Sánchez et al., 2018). Los supuestos adicionales de que el ruido es blanco y gaussiano implican que el ruido no está correlacionado a lo largo del tiempo y que su amplitud puede modelarse con precisión utilizando una media y una covarianza (es decir, el ruido se describe completamente por sus momentos primero y segundo) (Bradski, 2008).

Procedimiento de modelado-estimación

La figura 3 muestra un proceso iterativo del filtro de Kalman donde se muestra que el modelo se va ajustando con base en el tiempo y las nuevas medidas que se van tomando en el tiempo para hacer el ajuste y la corrección necesaria, como se describe a continuación.

Figura 3 Proceso iterativo del filtro de Kalman. Fuente: (Bradski, 2008)



En un instante t_1 , se determina la posición de un objeto p_1 . Sin embargo, debido a las imprecisiones inherentes del dispositivo de medición (como los cambios en la intensidad de la luz) el resultado de las mediciones es algo incierto. Por lo tanto, se determina que la precisión es tal que la desviación estándar involucrada es σ_1 (una sola variable). Por lo tanto, se puede establecer la probabilidad condicional, el valor en el instante t_1 , condicionado al valor observado de la medición σ_1 ; es decir, se tiene la probabilidad de que p tenga un valor, con base en la medición realizada. En este momento, se estima mejor $p_1 = p_1$ y la varianza $\sigma_{p1}^2 = \sigma_{p1}^2$.

Posteriormente, se realiza otra medición basada en cualquier opción de segmentación de objetos presentados en la sección anterior en el tiempo t_2 , obteniendo así p_2 con una varianza σ_{p2}^2 (que se supone menor que la primera). Para combinar estas mediciones y obtener una nueva con su propia variación (distribución gaussiana), se utilizan las ecuaciones 2 y 3, donde se observa que el nuevo valor es simplemente una combinación ponderada de las dos medias, y la ponderación se determina por las incertidumbres relativas de ambas mediciones (media condicional). El peso en estas ecuaciones se puede expresar como: si σ_{p1}^2 es mayor que σ_{p2}^2 (es decir, mayor variabilidad), entonces σ_{p2}^2 tendría mayor peso porque tiene menor variabilidad. Además, la incertidumbre en la estimación del nuevo p se ha reducido al combinar ambos datos (Maybeck, 9179).

Ahora que se sabe cómo obtener la siguiente medida, es posible continuar con este proceso N veces (N mediciones). Esto se debe a que se pueden combinar las dos

Ecuación 2. Estimación de la incertidumbre de la posición

$$p_{12} = \pi = \left(\frac{\sigma_{p_2}^2}{\sigma_{p_1}^2 + \sigma_{p_2}^2} \right) p_1 + \left(\frac{\sigma_{p_1}^2}{\sigma_{p_1}^2 + \sigma_{p_2}^2} \right) p_2$$

Ecuación 3. Estimación de la incertidumbre de la varianza

$$\sigma_{12}^2 = \frac{\sigma_{p_1}^2 \sigma_{p_2}^2}{\sigma_{p_1}^2 + \sigma_{p_2}^2}$$

primeras, luego la tercera con la combinación de las dos primeras, la cuarta con la combinación de las tres primeras, y así sucesivamente (Sánchez et al., 2018). Esto es lo que ocurre cuando se rastrea (o da seguimiento) una posición p a lo largo del tiempo: se obtiene una medida seguida de otra.

Generalmente, la ecuación 2 se reescribe como la ecuación 4 y la ecuación 3 como la ecuación 5, porque en estas nuevas formas es posible separar la información antigua de la nueva. La nueva información ($p_2 - p_1$) se denomina innovación.

Ecuación 4. Estimación de la nueva posición p

$$\widehat{p}_2 = p_1 + \frac{\sigma_{p_1}^2}{\sigma_{p_1}^2 + \sigma_{p_2}^2} (p_2 - p_1)$$

Ecuación 5. Cálculo de la varianza nueva

$$\widehat{\sigma}_2^2 = \left(1 - \frac{\sigma_{p_1}^2}{\sigma_{p_1}^2 + \sigma_{p_2}^2} \right) \sigma_{p_1}^2$$

Finalmente, la ecuación 6 muestra el factor de actualización iterativo óptimo, conocido como ganancia de actualización K ; así se obtiene la forma recursiva descrita en las ecuaciones 7 y 8. Para una explicación más detallada, se sugiere al lector consultar a Maybeck (1979).

Ecuación 6. Estimación de ganancia de actualización K

$$K = \frac{\sigma_{p_1}^2}{\sigma_{p_1}^2 + \sigma_{p_2}^2}$$

Ecuación 7. Estimación de la nueva posición p utilizando la constante K

$$\widehat{p}_2 = p_1 + K(p_2 - p_1)$$

Ecuación 8. Cálculo de la varianza nueva utilizando la constante K

$$\widehat{\sigma}_2^2 = (1 - K)\sigma_{p_1}^2$$

Resultados

A continuación se muestra un ejemplo de la metodología presentada. En la figura 4 se muestra un ejemplo de la identificación de vectores de flujo óptico por el método un poco denso de Lucas y Kanade. En la figura 4.(a) se muestra un ambiente donde un automóvil va en movimiento de izquierda a derecha y un peatón caminando en la misma dirección atrás del automóvil. Todo el paisaje es estático excepto esos dos objetos. Como se puede observar, el automóvil tiene mucha textura por lo que es más fácil identificar las esquinas o buenos puntos para seguir, y se muestran los vectores en la magnitud y dirección donde fue el movimiento de un tiempo t a un tiempo $t+1$. En la figura 4 (b) se extraen los vectores y se agrupan por el procedimiento mostrado en el módulo de segmentación donde el vector de la cabeza del peatón tiene diferente dirección y magnitud respecto del resto de los vectores que representan al automóvil.

Figura 4. Vectores resultantes de un movimiento a través de un proceso de flujo óptico



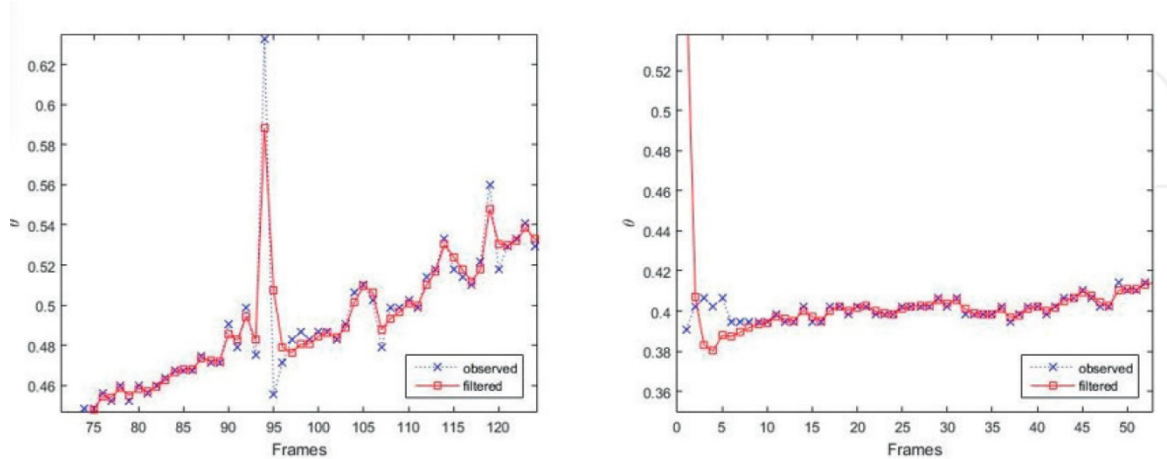
En la figura 5 se muestra un ejemplo de la generación de una cubierta convexa en un color rojo. Los puntos que delimitan a esta cubierta son los puntos donde se originó el movimiento. También se puede observar que dado que del peatón solo se obtuvo un vector de la cabeza, no se puede generar una cubierta convexa debido a que, por lo menos, se necesitan tres vectores.

Figura 5. Ejemplo de una cubierta convexa a partir de vectores de flujo óptico



Finalmente la figura 6 muestra un ejemplo del modelo de filtro de Kalman en el seguimiento del movimiento del objeto segmentado. En el primer gráfico se hace un acercamiento para identificar que si hay un dato atípico entonces el modelo tiene un error muy grande pero conforme pasa el tiempo se vuelve a ajustar acercándose a los datos azules que son las medidas observadas. En la segunda imagen se observa cómo al principio el error (es decir la diferencia entre lo observado y lo filtrado) es muy grande, pero a medida que pasa el tiempo la variabilidad y el error se va reduciendo.

Figura 6. Ejemplo de estimaciones y mediciones de filtro de Kalman



Discusión

Si bien en este trabajo se presentan algunas alternativas para los módulos de segmentación de objetos, modelado y predicción, se reconoce que las alternativas propuestas no son las únicas.

Para el caso del módulo de modelado segmentación, si bien comúnmente se hace por color, textura o forma hoy en día existen otros descriptores clásicos, tales como BRIEF (Binary Robust Independent Elementary Features), SIFT (Scale-Invariant Feature Transform) o SURF (Speeded Up Robust Features). Además, descriptores más recientes como FREAK (Fast Retina Keypoint), SIFT, SURF o basados en aprendizaje profundo, tales como Deep Descriptors (CNN-based), NetVLAD, o DELF (Deep Local Features) que son más robustos, pero costosos computacionalmente. Sin embargo, estos descriptores que pueden identificar puntos clave en la imagen, deben contar con información a priori de los objetos que se planean seguir.

Para el módulo de modelado, además de usar el filtro de Kalman es posible utilizar otras alternativas de modelado como una regresión lineal simple o cuadrática que permita ajustarse al tipo de movimiento en cuestión. Además diferentes tipos de modelado basado en identificación de sistemas (Sánchez et al., 2018) puede utilizarse como un enfoque particular de un sistema variante en el tiempo.

Conclusiones

En este capítulo se presenta una propuesta metodológica que demuestra ser una solución robusta y flexible para el seguimiento y modelado de objetos en movimiento en entornos visuales dinámicos. A través del uso de técnicas como segmentación por color, forma o textura, es posible extraer objetos relevantes de una escena. El análisis de flujo óptico complementa este proceso, permitiendo detectar cambios de posición entre imágenes consecutivas y superar limitaciones comunes como el problema de la apertura mediante la detección de esquinas. La agrupación de vectores y la construcción de cubiertas convexas permiten representar objetos completos, incluso cuando su forma cambia con el tiempo.

La incorporación del filtro de Kalman permite predecir de forma eficiente la trayectoria futura de los objetos, integrando mediciones sucesivas y reduciendo el impacto de posibles errores o ruidos. Esta metodología es especialmente útil en contextos donde se requiere tomar decisiones en tiempo real, como la detección de comportamientos anómalos en videovigilancia o el seguimiento preciso en sistemas autónomos. Además, su estructura modular permite ajustarse a distintos escenarios, optimizando su desempeño según las características específicas del entorno. En conjunto, este enfoque mejora significativamente la capacidad de comprensión visual de las máquinas y su interacción con el mundo real.

Trabajo futuro

El trabajo futuro de este capítulo se presenta en la forma de experimentar con diferentes alternativas para conocer cuál se ajusta mejor a las características de las escenas. Por ejemplo, si la escena varía mucho de iluminación, entonces una segmentación por color no es viable, pero una segmentación basada en descriptores puede ser muy útil.

Además, cualquier mecanismo de modelado que se escoja seguramente tienen parámetros que pueden hacer más preciso el modelo, aunque se puede caer en sobreajuste. Una exploración para cualquier modelo es la optimización de sus hiperparámetros que puede ser realizada con alguna metaheurística bioinspirada o no-bioinspirada, que permitan seleccionar los mejores valores de los parámetros de cada modelo.

Bibliografía

- Ballard, D. H. (1981). Generalizing the Hough transform to detect arbitrary shapes. *Pattern recognition*, 13(2), 111-122.
- Bradski, G., & Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc."
- Bitam, A., & Ameer, S. (2013, October). Multispectral image segmentation using Gabor filtering and local homogeneity analysis with application to MSG. In *3rd International Conference on Systems and Control* (pp. 838-843). IEEE.
- Farrokhnia, F., & Jain, A. K. (1991, January). A multi-channel filtering approach to texture segmentation. In *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 364-365). IEEE Computer Society.
- Fleyeh, H. (2004, December). Color detection and segmentation for road and traffic signs. In *IEEE Conference on Cybernetics and Intelligent Systems, 2004*. (Vol. 2, pp. 809-814). IEEE.
- Fuh, C. S., Cho, S. W., & Essig, K. (2000). Hierarchical color image region segmentation for content-based image retrieval system. *IEEE transactions on image processing*, 9(1), 156-162.
- Garcia-Ugarriza, L., Saber, E., Amuso, V., Shaw, M., & Bhaskar, R. (2008, March). Automatic color image segmentation by dynamic region growth and multimodal merging of color and texture information. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 961-964). IEEE.
- Garje, P. D., Nagmode, M. S., & Davakhar, K. C. (2018, February). Optical flow based violence detection in video surveillance. In *2018 international conference on advances in communication and computing technology (ICACCT)* (pp. 208-212). IEEE.
- Harris, C., & Stephens, M. (1988, August). A combined corner and edge detector. In *Alvey vision conference* (Vol. 15, No. 50, pp. 10-5244).
- Horn, B. K., & Schunck, B. G. (1981). Determining optical flow. *Artificial intelligence*, 17(1-3), 185-203.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.
- Littmann, E., & Ritter, H. (1997). Adaptive color segmentation-a comparison of neural and statistical methods. *IEEE Transactions on neural networks*, 8(1), 175-185.
- Parthasarathy, P., & Vivekanandan, S. (2019, March). Detection of suspicious human activity based on CNN-DBNN algorithm for video surveillance applications. In *2019 Innovations in Power and Advanced Computing Technologies (i-PACT)* (Vol. 1, pp. 1-7). IEEE.
- Leavers, V. F. (1993). Which hough transform?. *CVGIP: Image understanding*, 58(2), 250-264.
- Liu, C., Ying, J., Han, F., & Ruan, M. (2018, July). Abnormal human activity recognition using bayes classifier and convolutional neural network. In *2018 IEEE 3rd international conference on signal and image processing (ICSIP)* (pp. 33-37).

IEEE.

- Lu, C. S., Chung, P. C., & Chen, C. F. (1997). Unsupervised texture segmentation via wavelet transform. *Pattern recognition*, 30(5), 729-742.
- Lucas, B. D., & Kanade, T. (1981, August). An iterative image registration technique with an application to stereo vision. In *IJCAI'81: 7th international joint conference on Artificial intelligence* (Vol. 2, pp. 674-679).
- Maybeck P. Introduction. In: *Stochastic Models, Estimation and Control*. 1st ed. London: Academic Press; 1979. p. 1-16
- Sanchez-Garcia, A. J., Rios-Figueroa, H. V., Marin-Hernandez, A., & Acosta-Mesa, H. G. (2014, February). Tracking and prediction of motion of segmented regions using the Kalman filter. In *2014 International Conference on Electronics, Communications and Computers (CONIELECOMP)* (pp. 88-93). IEEE.
- Sánchez, A., Ríos, H., Quintana, G., & Marín, A. (2018). Predicting Collisions in Mobile Robot Navigation by Kalman Filter. *Kalman Filters: Theory for Advanced Applications*, 151.
- Sanchez-Garcia, A. J., Rios-Figueroa, H. V., Garnier, H., Quintana-Carapia, G., Rechy-Ramirez, E. J., & Marin-Hernandez, A. (2018). Predicting collisions: time-to-contact forecasting based on probabilistic segmentation and system identification. *Advanced Robotics*, 32(8), 426-442.
- Sarcar, S. T., & Yousuf, M. A. (2021, December). Detecting violent arm movements using cnn-lstm. In *2021 5th International Conference on Electrical Information and Communication Technology (EICT)* (pp. 1-6). IEEE.
- Shi, J., & Tomasi, C. (1994). Good features to track. *9th IEEE conference on computer vision and pattern recognition*.
- Singh, K., Preethi, K. Y., Sai, K. V., & Modi, C. N. (2018, December). Designing an efficient framework for violence detection in sensitive areas using computer vision and machine learning techniques. In *2018 Tenth International Conference on Advanced Computing (ICoAC)* (pp. 74-79). IEEE.
- Wang, M., Han, G., Tu, Y., Chen, G., & Gao, Y. (2008, December). Unsupervised texture Image segmentation based on Gabor wavelet and multi-PCNN. In *2008 Second International Symposium on Intelligent Information Technology Application* (Vol. 2, pp. 376-381). IEEE.
- Wu, Y., Liu, Q., & Huang, T. S. (2000, January). An adaptive self-organizing color segmentation algorithm with application to robust real-time human hand localization. In *Proceedings of Asian Conference on Computer Vision* (pp. 1106-1111).