ISSN 1870-4069

Implementación de algoritmo de pruebas para la detección de comportamientos humanos utilizando el filtro de Kalman para inferencia de actividades mediante el uso de machine learning

Braulio Israel Alejo-Cerezo, Juan Pablo Pérez-Monje,
Selene Ramírez-Rosales, Alvaro Anzueto-Ríos,
Jorge Luis Pérez-Ramos

Universidad Autónoma de Querétaro,
Facultad de Informática,
México

{bralioiac, juanpabloperez0299, seleneramirezrosales}@gmail.com,
aanzuetor@ipn.mx, jorge.luis.perez@uaq.edu.mx

Resumen. Los sistemas de videovigilancia son herramientas tecnológicas que ayudan al ser humano a la monitorización de áreas de interés para fines de seguridad, supervisión y prevención de delitos. Estudios indican que los sistemas de videovigilancia no supervisados exhiben una ventaja al no poseer limitaciones humanas, como parte de ello, el objetivo principal de este trabajo recae en la implementación de un sistema de videovigilancia no supervisada, enfocado en la detección de comportamientos humanos asociados con actos delictivos, utilizando algoritmos de identificación y seguimiento de personas en apoyo con el filtro de Kalman, implementando finalmente una respuesta de alarma y notificación al usuario. Con ello, se obtuvieron los resultados del análisis de vídeos cuyas escenas fueron capturadas en diferentes entornos, validando así la efectividad del sistema para la búsqueda e identificación de comportamientos.

Palabras clave: Videovigilancia, patrones, comportamientos, detección, seguimiento, vídeos.

Implementation of Test Algorithm for Human Behaviors Detection Using the Kalman Filter for Activities Inference Using Machine Learning

Abstract. Surveillance systems are technological tools that help humans in monitoring regions of interest for security, supervision, and crime prevention tasks. Researchers say unsupervised surveillance systems are advantageous since they do not depend on human limitations. Therefore, the main objective of this document is the implementation of an unsupervised surveillance system focused on the detection of human behaviors related to criminal acts, all of this by using algorithms for people detection and tracking, and finally, implementing an alarm

response and user notification system. This system lets us analyze videos in which scenes were recorded with different characteristics, validating the system's effectiveness in searching and identifying behaviors.

Keywords: Surveillance, patterns, behaviours, detection, tracing, videos.

1. Introducción

Desde su aparición, los sistemas de videovigilancia han conformado un conjunto de herramientas esenciales para el resguardo de espacios públicos y privados, tal como lo son centros comerciales, bancos, instituciones gubernamentales, escuelas, negocios, casas, etc. En la actualidad estos sistemas han evolucionado con herramientas propias de Inteligencia Artificial, principalmente enfocadas a la reducción de las principales fuentes humanas de inefectividad en relación con las cámaras de seguridad [2].

Un sistema típico de videovigilancia consta de cinco partes: detección de objetos, clasificación de objetos, seguimiento de objetos, entendimiento y descripción de comportamientos, y la identificación [7]. Las cámaras empleadas en videovigilancia graban días enteros de actividades, lo cual resulta en una gran cantidad de datos de vídeo que hacen de los procesos de búsqueda una tarea laboriosa y altamente demandante para un observador humano [8], cuando pretende obtener información relevante.

Estos sistemas cuentan con una persona encargada de evaluar los movimientos registrados en cada escena, de acuerdo a sus conocimientos y criterio de cada situación. Sin embargo, analizar múltiples cámaras de manera simultánea como sistemas distribuidos aumenta su complejidad y genera dependencia a un operador en la detección de actos delictivos, ya que es necesario monitorizar múltiples pantallas de forma simultánea y sincronizada, prestando atención a los detalles de cada escena de forma continua.

Lo anterior presenta limitaciones en la efectividad de la monitorización debido al error humano, como ceguera no intencional a causa de errores por falta de atención [1, 9]. Se reporta que en México, la tasa de incidencia delictiva ha presentado un crecimiento en el delito de robo a casa habitación (rch), esto de acuerdo con cifras de la Encuesta Nacional de Victimización y Percepción sobre Seguridad Pública (ENVIPE).

Actualmente, esta cifra apunta a 1,880 delitos por rch$\times$ 100,000 hab, no obstante, la [6] reporta que, debido a la contingencia sanitaria generada por el virus SARS-CoV2 (causante de la COVID-19), a partir del levantamiento de la ENVIPE 2020, con año de referencia 2019, se han registrado cambios estadísticamente significativos con respecto a los ejercicios anteriores, presentando valores más bajos de lo normal.

A pesar de ello, el número de personas que pueden verse beneficiadas mediante el presente proyecto se puede conocer gracias a que la ENVIPE 2021 consideró una tasa de 1.3 delitos por víctima durante el 2020, proporción que, de aplicarse al número total de delitos por robo a casa habitación (1.6836 millones de delitos), da como resultado un total de 1.295 millones de víctimas.

Este mismo estudio permite estimar el costo total a consecuencia de la inseguridad por delito, y estima que, sólo en el año de 2020, el delito en hogares alcanzó un monto

de pérdidas aproximado de 277.6 mil mdp. Para ilustrar la gravedad de esta cifra, se estima que la misma representa 1.85 % del PIB [6].

El beneficio de proyectos proyectos enfocados a sistemas de vigilancia se fundamenta en la investigación realizada por [3], donde establece que la tecnología es una alternativa efectiva si se utiliza como técnica de prevención situacional al delito, esto quiere decir que los dispositivos tecnológicos implementados en sistemas de vigilancia influyen directamente en los riesgos que perciben los infractores, y por ello, inhibe la comisión del delito. Por lo anterior, se propone el desarrollo de un sistema de vigilancia para la detección de comportamientos humanos utilizando el filtro de Kalman para inferencia de actividades mediante el uso de Machine Learning.

2. Marco teórico

Esta sección explica los fundamentos teóricos en los que se basa la presente investigación. En primer lugar se tiene el Procesamiento Digital de Imágenes, que es el proceso por el cual una máquina puede interpretar la información de los elementos que conforman una imagen, y es ampliamente usado en campos como la medicina, física, arqueología, etc. Por otro lado, el filtro de Kalman es un predictor lineal, el cual es empleado en sistemas no lineales para obtener la dinámica del sistema y el movimiento de los objetos en la escena.

2.1. Procesamiento digital de imágenes

Una imagen digital puede definirse como una función de dos dimensiones $f(x, y)$ donde x y y son coordenadas espaciales de un plano, y la amplitud de f en cualquier par de coordenadas es llamado intensidad o nivel de gris de la imagen en ese punto [5], sin embargo, existen también representaciones de imágenes en donde cada píxel de color es una combinación de los colores (también llamados canales) rojo, verde y azul [11]. Si una coordenada (x,y), así como el valor de f son cantidades finitas y discretas, la función toma el nombre de imagen digital.

El procesamiento de imágenes hace referencia al procesamiento de imágenes digitales empleando equipos de cómputo y toma en consideración las técnicas utilizadas para desarrollar dicho procesamiento, cuyo fin será obtener una mejora en la imagen o extraer información que resulte útil para el propósito que se persigue [12]. Una imagen digital se compone de un número finito de elementos, cada uno con un valor y ubicación particular, estos elementos son llamados elementos de imagen o píxeles.

2.2. Filtro de Kalman

El filtro de Kalman es un conjunto de ecuaciones matemáticas que son capaces de proveer un método computacional recursivo eficiente para estimar el estado de un proceso, minimizando el error cuadrático medio [14, 10].

El filtro de Kalman aborda el problema de estimar el estado $x \in \mathbb{R}^n$ de un proceso discreto controlado que está definido por la siguiente ecuación diferencial estocástica [13]:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}. \tag{1}$$

En un tiempo k, con una medición $z \in \mathbb{R}^m$:

$$z_k = Hx_k + v_k, \tag{2}$$

donde w_k y v_k son variables aleatorias que representan el ruido del proceso y la medición del ruido respectivamente, y se asume que son independientes una de la otra, con distribución normal de probabilidad:

$$\begin{aligned} p(w) &\sim N(0, Q), \\ p(v) &\sim N(0, R), \end{aligned} \tag{3}$$

donde Q es covarianza del ruido del proceso y R la covarianza del ruido de la medición, las cuales se asumen constantes. La matriz A con dimensión $n \times n$ relaciona el estado x en el tiempo $k-1$ con el estado x en el tiempo actual k. La matriz B de dimensión $n \times l$ relaciona una entrada opcional $u \in \mathbb{R}^l$ al estado x.

Por último, la matriz H $m \times n$ en la medición relaciona el estado con la medición z_k. Empleando las ecuaciones anteriores, así como mediciones reales $\hat{z}_k$ en cada tiempo k, el filtro de Kalman se usa para estimar de forma recursiva la media $\hat{x}_k$ y el error de covarianza P_k. El filtro se aplica en dos pasos: actualización del tiempo:

$$\hat{x}_k = A\hat{x}_{k-1} + Bu_{k-1}, \tag{4}$$

$$P_k = AP_{k-1}A^T + Q, \tag{5}$$

Y actualización de la medición:

$$K = p_k^- H^T \left(HP_k^- H^T + R\right)^{-1}, \tag{6}$$

$$\hat{x}_k = \hat{x}_k^- + K\left(\hat{x}_k - H\hat{x}_k^-\right). \tag{7}$$

Cabe mencionar que el filtro de Kalman es óptimo ya que la matriz de ganancia del filtro K minimiza el error de la covarianza en el seguimiento de la iteración siguiente.

3. Implementación de la detección de comportamientos

La propuesta de este proyecto se aborda mediante la Figura 1, contando con cuatro fases correspondientes a: 1) el sensor de entrada de datos, 2) la multimedia extraída, 3) el sistema de procesamiento de datos y 4) la salida del sistema. Con ello, se planteó la programación de las funciones necesarias para identificar a cada persona dentro de las escenas y establecer el seguimiento de las mismas, evitando el conflicto de detección de sujetos mencionado por [4] mediante la implementación del filtro de Kalman.

Lo anterior debido a su mayor afinidad con el problema abordado en comparación con los equivalentes filtros Bayesianos, ya que estos se enfocan principalmente en estimar una función de densidad probabilística de los estados observados en el tiempo,

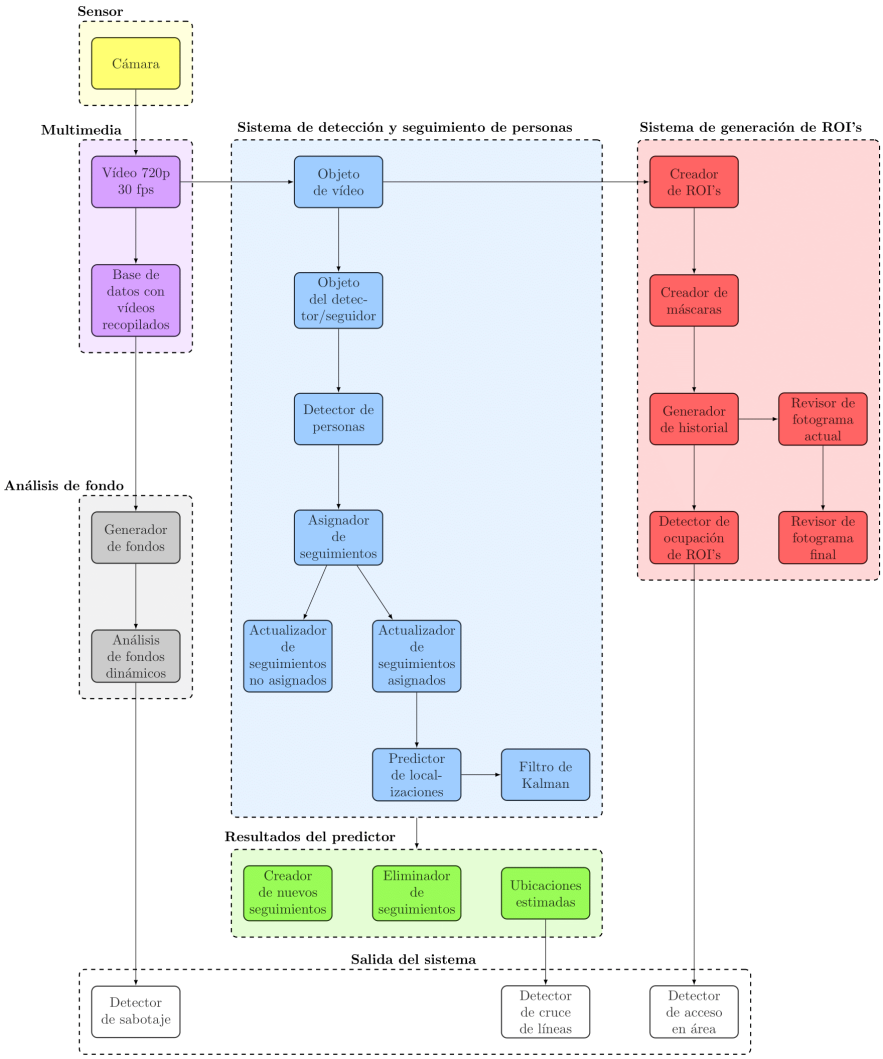


Fig. 1. Arquitectura general del sistema.

por ello, se encuentran generalmente dirigidos a estimar estados en el campo de la robótica, en donde se pretende determinar el estado actual del sistema dado un cierto historial de entradas y observaciones.

Por el contrario, el filtro de Kalman, como un caso especial de los filtros de Bayes, se utilizó para realizar una suposición en un sistema dinámico con información incierta, y establecer, a partir de ésta, el estado siguiente de dicho sistema, almacenando únicamente el estado previo.

Esto permitió realizar el seguimiento del movimiento de los sujetos basando este mismo en el punto central de cada persona identificada, utilizando el centroide de cada seguimiento para predecir su estado siguiente. Una vez hecho esto, se realizó la

Escena A1: pasillo, sin obstáculos, con puerta única ubicada a la izquierda de la escena.

Escena A2: pasillo y jardín, obstáculos por árboles y arbustos, con dos puertas.

Escena A3: pasillo, obstáculo de castillo cilíndrico a la izquierda, sin puertas.

Fig. 2. Vista y descripción de las escenas utilizadas en el proceso de validación.

actualización de cada caja tomando el largo y ancho actual para también establecer su tamaño futuro, permitiendo así corregir adecuadamente las localizaciones estimadas.

Por último, en este proceso se otorgó un número de identificación específico a cada caja para identificar su trayectoria de paso. A partir de ello, se buscó automatizar el proceso de seguimiento y predicción de los desplazamientos de cada sujeto en la escena, para no depender de un operador humano que, de acuerdo con [8, 15], genera una baja efectividad en la prevención de crímenes tras el monitoreo constante por largos periodos de tiempo.

Lo anterior con el fin de reconocer los siguientes tres comportamientos humanos: con cruce de líneas, acceso a área definida por el usuario y sabotaje de cámaras. Validando su eficacia en pruebas que utilizaron una base de datos de vídeos captados de manera propia, los cuales se observan en la Figura 2.

De acuerdo con lo establecido anteriormente, como primer paso, se buscó implementar un sistema dedicado al reconocimiento y seguimiento de los comportamientos humanos relacionados con actos delictivos, mencionados previamente, basado en el número de fotogramas por segundo y la secuencia de aparición de cada sujeto en escena. Posteriormente se formuló un modelo de detección de vulnerabilidades, registrando el cruce de líneas arbitrarias en la escena, ingreso a zonas determinadas en la misma y la obstrucción del campo de visión de la cámara.

En lo que respecta a estas dos primeras fases, se utilizaron grabaciones de videovigilancia obtenidas mediante datasets públicos de uso académico, y, posteriormente, fueron obtenidas las grabaciones de videovigilancia correspondientes

Tabla 1. Tabla de descripción de vídeos obtenidos.

Escena	Procedencia	Duración
A1	Nikon D3500	30 min
A2	Poco X3	20 min
A3	FOSCAM FI9804W	120 min
A4	rawi_mages_pedestrians[1]	10 min
A5	Virat_s_000102[2]	20 min

a instituciones públicas empleando los instrumentos referidos en la Tabla 1, donde se concentró el total de escenas utilizadas para esta investigación.

En lo referente al algoritmo de detección de cruce de líneas, este se realizó mediante el trazado de objetos llamados regiones de interés (del inglés Regions of Interest o ROI's), los cuales cumplieron el papel de interpretar límites en la escena dibujados por el usuario utilizando la función drawline.

Este método de trazado de líneas arbitrarias otorgó al programa la versatilidad para utilizar dichas líneas de acuerdo con las necesidades de monitorización de la escena, pues pueden ser empleadas tanto como delimitadores en la misma, detección de ingreso o salida de personas, o incluso como guías de trayectorias para el comportamiento deseado en los individuos que ocupen la zona.

El trazado de la región correspondiente al área restringida fue realizado utilizando la misma metodología empleada en el algoritmo de líneas previamente descrito, con la diferencia de que, para este caso, la ROI fue dibujada mediante la función drawpolygon, la cual permitió establecer un conjunto de puntos indefinido hasta que el usuario finalmente decidiera cerrar el polígono en cuestión.

De igual manera, la implementación de ambos algoritmos en el objeto de vídeo permitió establecer un conjunto de máscaras, con el fin de realizar un seguimiento de la región y líneas establecidas por el usuario durante el análisis del vídeo.

Una vez definidas las máscaras individuales de cada objeto, estas fueron conjuntadas en una única máscara general, con el tamaño de la escena estudiada, para poder ser comparadas con las bounding boxes pertenecientes a las personas, en cada uno de los fotogramas siguientes.

La interacción entre bounding boxes y ROI's se generó a través de la función inROI, con la cual, todos los píxeles pertenecientes a la máscara fueron comparados con las coordenadas delimitantes de cada bounding box, dando como resultado la interpretación de cuáles sujetos, de todos aquellos identificados en la escena, se encontraban interactuando con el área o lineas trazadas por el usuario. Una vez que esta interacción fue identificada por el programa, se realizó un primer registro en un vector

[1] E. Gebhardt and M. Wolf, "CAMEL Dataset for Visual and Thermal Infrared Multiple Object Detection and Tracking," IEEEInternational Conference on Advanced Video and Signal-based Surveillance (AVSS), 2018.

[2] "A Large-scale Benchmark Dataset for Event Recognition in Surveillance Video" by Sangmin Oh, Anthony Hoogs, Amitha Perera, Naresh Cuntoor, Chia-Chih Chen, Jong Taek Lee, Saurajit Mukherjee, J.K. Aggarwal, Hyungtae Lee, Larry Davis, Eran Swears, Xiaoyang Wang, Qiang Ji, Kishore Reddy, Mubarak Shah, Carl Vondrick, Hamed Pirsiavash, Deva Ramanan, Jenny Yuen, Antonio Torralba, Bi Song, Anesco Fong, Amit Roy-Chowdhury, and Mita Desai, in Proceedings of IEEE Comptuer Vision and Pattern Recognition (CVPR), 2011.

Tabla 2. Tabla de descripción de vídeos obtenidos.

Nivel	Descripción de alerta
1	Alerta de notificación de evento de bajo riesgo
2	Alerta preventiva a un evento de alto riesgo
3	Alerta de notificación de evento de alto riesgo

temporal llamado historial, que se encargó de almacenar el comportamiento detectado el cual puede ser cruce de línea 1, cruce de línea 2 o acceso al área restringida, además de que almacenaba información adicional como el fotograma de inicio y fin.

Con lo anterior se generó un listado de los eventos detectados en el orden en que transcurren en cada una de las grabaciones, los cuales fueron interpretados de forma semántica para comunicar al usuario cada uno de los comportamientos detectados. El algoritmo correspondiente a la detección del comportamiento de sabotaje se implementó por medio de la generación de fondos dinámicos, los cuales representaron el promedio de los fotogramas en intervalos de cinco segundos.

Cada vez que se obtuvo un nuevo fondo, este fue comparado con el promedio de los anteriores mediante una correlación de los histogramas de cada uno, de esta manera, cuando se mantuvo un coeficiente de correlación por encima del 80 % o 90 %, el nuevo fondo fue promediado junto a los demás.

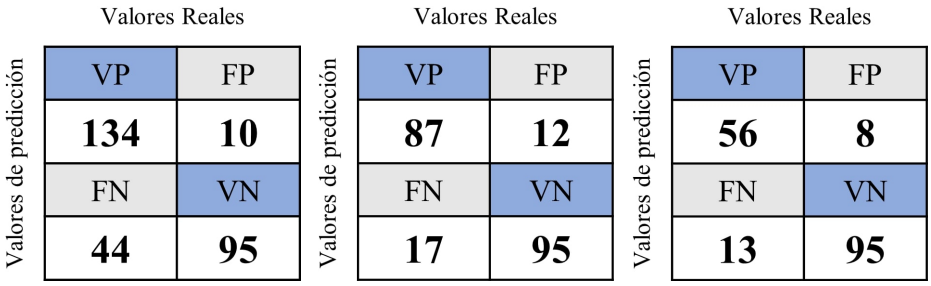
Por el contrario, cuando la correlación se encontró en un valor por debajo del 40 % o 30 % se optó por notificar al usuario acerca de sabotaje, ya que el algoritmo interpretó que un cambio radical en el último fondo obtenido pertenecía a una modificación de alto riesgo para la monitorización de la escena. Es importante mencionar que el umbral para evaluar el coeficiente de correlación fue ajustado a los valores que el usuario definió de acuerdo con el tipo y características de la escena estudiada.

Las salidas del algoritmo fueron validadas mediante un sistema de evaluación de umbrales, con el cual se establecieron opciones de respuesta de acuerdo a los parámetros de: tipo de comportamiento detectado, duración de comportamiento (para el caso de detección de acceso al área restringida), e incidencia de comportamiento (para el caso de la detección de cruce de líneas), con ello el sistema fue capaz de presentar una respuesta particular ante cada evento identificado, y de acuerdo con las características del mismo, obtuvo como salida la notificación al usuario y/o la activación de la alarma de seguridad, así como también la asignación de un nivel de alerta correspondiente a la situación encontrada con el fin de informar al usuario la gravedad del propio evento, los niveles de alerta son descritos en la Tabla 2.

Respecto al medio de notificación al usuario, se estableció un sistema de envío de correos electrónicos, para lo cual, se adaptó una plantilla con la estructura inicial de un email, considerando una sintaxis basada en el formato html para el envío de la información de interés del evento detectado empleando un socket de Outlook por medio de funciones propias de la plataforma.

4. Resultados

El análisis de cada una de las escenas descritas previamente se llevó a cabo por medio de matrices de confusión. La Figura 3 indica que la escena A1 cuenta con un



(a) Matriz de confusión para la escena A1.

(b) Matriz de confusión para la escena A2.

(c) Matriz de confusión para la escena A3.

Fig. 3. Resultados del reconocimiento de comportamientos en las escenas de validación.

Tabla 3. Tabla comparativa sobre las métricas de desempeño.

Fórmula	**A1**	**A2**	**A3**
$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$	0.7528	0.8365	0.8115
$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$	0.9305	0.8787	0.8750

total de 178 eventos, de los cuales 134 fueron identificados con éxito, 44 no pudieron ser identificados por el sistema y 10 se detectaron a pesar de no aparecer realmente en el vídeo. Por otra parte A2 registró un total de 104 eventos, con 87 detectados adecuadamente, 17 no reconocidos y 12 eventos detectados aunque inexistentes. Por último, A3 registró 69 eventos de los cuales detectó correctamente 56, pasando por alto únicamente 13 y reportando 8 eventos de más.

4.1. Desempeño del clasificador

Con los datos de las matrices de confusión mostrados en la figura previa se obtuvieron las métricas mostradas en la Tabla 3 de True Positive Rate y Positive Predictive Value. La información obtenida mediante las matrices de confusión y las métricas de desempeño, fue utilizada para generar un único gráfico de tabla ROC, con el cual comparar de manera visual la relación del TPR con el PPV.

La Figura 4 expone mediante la línea roja que la escena A1 posee un área bajo la curva correspondiente a la efectividad del 75.28 % en la detección de los eventos de interés (dado que la efectividad es mayor al 60 %, la escena revela un adecuado criterio de identificación de comportamientos), considerando igualmente que el número de eventos registrados es el mayor en comparación con los demás vídeos.

Por otro lado, la línea azul ilustra el hecho de que la escena A2 obtuvo el mayor porcentaje de efectividad encontrado en el sistema, con hasta un 83.65 % de éxito en el reconocimiento de los tres comportamientos. Finalmente, la escena A3 (línea verde), mostró un área bajo la curva relacionada con la efectividad del 81.15 %.

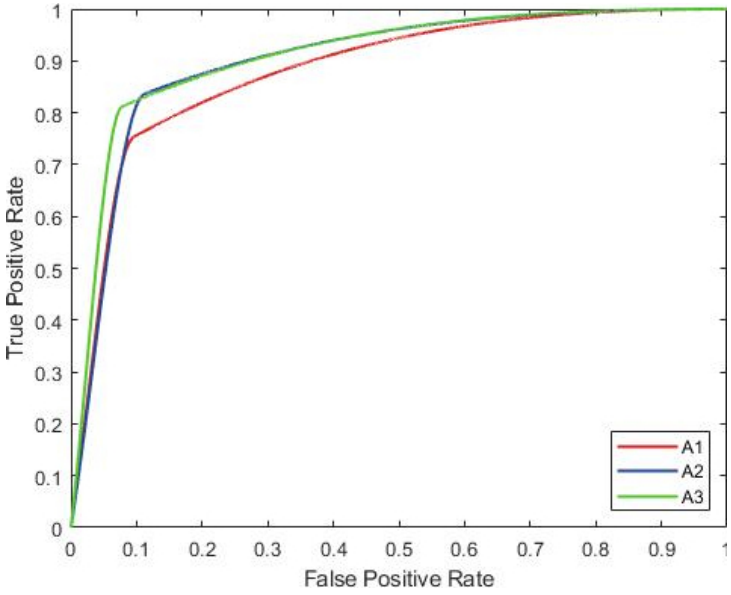


Fig. 4. Comparativa de las curvas ROC obtenidas en las escenas A1, A2 y A3.

5. Conclusiones

Una vez analizados los resultados obtenidos, se plantean las siguientes líneas de investigación que pueden ser abordadas a partir de este trabajo. En primer lugar, se propone la implementación de un algoritmo metaheurístico de optimización para la calibración autónoma del sistema, con el fin de encontrar los mejores parámetros del detector y seguidor de personas, así como umbral de correlación óptimo para la detección de sabotajes en cada escena.

En segundo lugar, se recomienda la búsqueda e implementación de nuevos comportamientos relacionados con actos delictivos, los cuales agreguen mayores opciones de identificación de eventos, tales como notificación por ausencia de sujetos, detección de dirección de flujo de personas y detección de merodeo.

Por otro lado, se propone realizar modificaciones al clasificador aumentando las características para generar un análisis de cada fotograma más allá de las interacciones de cajas de personas con ROI's y comparación de histogramas, mejorando la detección y el seguimiento de personas con la adición de datos biométricos almacenados en un sistema de perfiles, para clasificar a los sujetos de acuerdo con su potencial de amenaza en la escena.

En cuanto a las ROI's, se propone el desarrollo de un algoritmo de detección y generación de ROI's automático basado en el análisis de las regiones de mayor movimiento de la escena, con el fin de separarlas y delimitarlas mediante el algoritmo de segmentación de regiones Watershed, permitiendo establecer un estudio únicamente en las zonas de mayor actividad del vídeo.

Por último, se recomienda la adaptación de nuevos actuadores como parte de la respuesta física del prototipo, con el fin de permitirle tomar acciones específicas en

respuesta a los comportamientos detectados, como el encendido de luces inteligentes o el cierre de cerraduras automáticas, estableciendo un sistema adaptable en el área de la domótica.

Referencias

1. Bredemeier, K., Simons, D. J.: Working memory and inattentional blindness. Psychonomic Bulletin and Review, vol. 19, pp. 239–244 (2012) doi: 10.3758/s13423-011-0204-8
2. Chen, C., Surette, R., Shah, M.: Automated monitoring for security camera networks: Promise from computer vision labs. Security Journal, vol. 34, no. 3, pp. 389–409 (2020) doi: 10.1057/s41284-020-00230-w
3. Clarke, R. V.: Situational cime prevention: Successful case studies, Harrow and Heston, Publishers (1997)
4. Dan, X., Yan, Y., Ricci, E., Sebe, N.: Detecting anomalous events in videos by learning deep representations of appearance and motion. Comput Vis Image Underst, pp. 117–127 (2017) doi: 10.1016/j.cviu.2016.10.010
5. González, R. C., Woods, R. E.: Digital Image Proccessing, New York: Pearson (2018)
6. INEGI: Encuesta nacional de victimización y percepción sobre seguridad pública envipe 2021. (2021) https://www.inegi.org.mx/programas/envipe/2021/
7. Kong, L., Dai, R.: Object-detection-based video compression for wireless surveillance systems. IEEE MultiMedia, vol. 24, no. 2, pp. 76–85 (2017) doi: 10.1109/MMUL.2017.29
8. Meghdadi, A. H., Irani, P.: Interactive exploration of surveillance video through action shot summarization and trajectory visualization. In: IEEE Transactions on Visualization and Computer Graphics, vol. 19, pp. 2119–2128 (2013) doi: 10.1109/TVCG.2013.168
9. Sasse, M. A.: Not seeing the crime for the cameras? Communications of the ACM, vol. 53, no. 2, pp. 22–25 (2010) doi: 10.1145/1646353.1646363
10. Sorenson, H. W.: Least-squares estimation: from Gauss to Kalman. IEEE Spectrum, vol. 7, no. 7, pp. 63–68 (1970) doi: 10.1109/MSPEC.1970.5213471
11. Thanki, R. M., Kothari, A. M.: Digital Image Processing using SCILAB, Springer International Publishing (2019) doi: 10.1007/978-3-319-89533-8
12. Tyagi, V.: Understanding digital image processing, CRC Press (2018) doi: 10.1201/9781315123905
13. Welch, G., Bishop, G.: An introduction to the Kalman filter. (1995) perso.crans.org/club-krobot/doc/kalman.pdf
14. Welch, G. F.: Kalman filter. Computer Vision, Springer International Publishing, pp. 721–723 (2021) doi: 10.1007/978-3-030-63416-2_716
15. Wiliem, A., Madasu, V., Boles, W., Yarlagadda, P.: A suspicious behaviour detection using a context space model for smart surveillance systems. Computer Vision and Image Understanding, vol. 116, pp. 194–209 (2012) doi: 10.1016/j.cviu.2011.10.001