Análisis del comportamiento de clientes mediante técnicas de inteligencia artificial y visión por computadora

Salvador Apablaza[1], Carolina Sol Fernández[1], Nicolás Gabriel Locatti[1], Diego Durante[2], Sebastián Verrastro[1], Juan Carlos Gómez[2,3]

1 Universidad Tecnológica Nacional, Facultad Regional Buenos Aires, Av. Medrano 951, (C1179AAQ), Ciudad Autónoma de Buenos Aires, Argentina
2 Universidad Tecnológica Nacional, Facultad Regional Buenos Aires, Grupo de Inteligencia Artificial y Robótica (GIAR), Av. Medrano 951, (C1179AAQ) Ciudad Autónoma de Buenos Aires, Argentina
3 Instituto Nacional de Tecnología Industrial (INTI), Centro de Electrónica e Informática, Av. Gral. Paz 5445, (B1650WAB) San Martín, Buenos Aires, Argentina

carolinasolfernandez@gmail.com

Recibido el 30 de marzo de 2023, aprobado el 27 de abril de 2023

Resumen

En el presente trabajo se desarrolla un *framework* de detección de personas para analizar el comportamiento de los clientes en una tienda minorista mediante el procesamiento de videos de vigilancia. Para ello, se propone un *pipeline* compuesto por un preprocesamiento del video, un procesamiento con un modelo basado en redes neuronales, y un postprocesamiento orientado a correcciones. Se proveen indicadores útiles para los negocios, entre ellos, un mapa de calor que muestra los espacios ocupados. El resultadoes constatado con un etiquetado manual para evaluar el rendimiento del algoritmo. Se demuestra la viabilidad de este método para comprender el comportamiento de los clientes.

PALABRAS CLAVE: DETECCIÓN DE PERSONAS - VISIÓN POR COMPUTADORA - INTELIGENCIA ARTIFICIAL - MAPA DE CALOR - COMPORTAMIENTO DE CLIENTES

Abstract

In the present work, a framework for people detection is developed to analyze customer behavior in a retail store using surveillance videos. For this purpose, a three-stage pipeline consisting of video preprocessing, processing with a neural network-based model, and correction-oriented post-processing is proposed. The pipeline provides useful business indicators, including a heatmap showing occupied spaces. The results are verified through manual labeling to evaluate the algorithm's performance. The study concludes by demonstrating this method's viability for understanding customer behavior.

KEYWORDS: PEOPLE DETECTION - COMPUTER VISION - ARTIFICIAL INTELLIGENCE - HEATMAP – CUSTOMER BEHAVIOR

Introducción

En la actualidad, la mayoría de las soluciones de procesamiento de imágenes que buscan detectar la cantidad de peatones presentes en un fotograma (*frame*) se realizan en espacios y condiciones que no son tan desafiantes como las del mundo real, donde aparecen oclusiones entre objetos, las condiciones de iluminación no son las ideales o se fusionan con información de fondo (Ahmed *et al.*, 2021). Si se quisiera implementar uno de estos algoritmos para un área comercial, por ejemplo, con el objetivo de contar personas, además de los errores que podrían darse por cruces de personas, ocurrirían varios errores adicionales debido a la cantidad de obstáculos presentes en el espacio (mesas, góndolas, cartelería, etc.), dificultando la detección y empeorando el conteo de personas.

Según estudios recientes, al menos el 70 % de las personas están ocluidas en videos tomados en tiendas, bancos y estaciones de transporte (Ning *et al.*, 2021). Esto hace que los métodos tradicionales de detección de personas formados por un extractor de características y un clasificador hayan quedado en desuso frente a los métodos de detección basados en aprendizaje profundo (*deep learning*). Dentro de este segundo tipo de algoritmos, hay un abanico de opciones muy grande, tales como redes neuronales convolucionales, redes neuronales recurrentes, entre otros. Esto genera una dispersión muy grande tanto en el tiempo que tardan en ejecutarse los algoritmos como en los resultados obtenidos de los mismos.

El objetivo de esta investigación es mejorar el análisis de mercado del comercio físico tradicional, el cual ha perdido terreno frente a las tiendas y el comercio web en los últimos años. Según Chava *et al.* (2022), el comercio web ha aumentado su cuota de mercado del 0,63 % en 1999 al 13,3 % en 2021. Se busca generar herramientas para analizar a los clientes dentro de ambientes internos cerrados y obtener métricas de valor para los negocios.

La Nube

Una tecnología que podría dar atención a esta problemática, y que además está en auge es la Nube (*Cloud*) (Salmon y Parmar, 2022). Realizar un buen procesamiento local requiere de poder computacional en el hardware donde se está llevando a cabo el análisis, por lo que delegar esta responsabilidad a la nube es más que atractivo. La tendencia tecnológica de los últimos años, liderada por el incremento en las tasas de transferencias de datos y el espacio de almacenamiento, generan el contexto ideal para que esta tecnología se posicione como una de las más utilizadas en casi todos los ámbitos (Mane, 2022).

El inconveniente más grande de este acercamiento es el costo asociado a estos servicios. Como ha pasado a lo largo de la historia con tecnologías que están en proceso de maduración, para un análisis de este tipo el costo es sumamente elevado, por lo que se ha desestimado para este estudio.

Contribuciones

En este artículo se presenta un marco de trabajo (*framework*) de detección de personas diseñado para su implementación en entornos de bajo poder de procesamiento, como una computadora hogareña. Utilizando videos de una cámara de vigilancia ubicada en una librería, se obtienen indicadores de negocio tales como el momento de máximo aforo, las personas presentes en cada instante y un mapa de calor que muestra la ocupación del espacio por la clientela.

En el presente trabajo se valida, a su vez, el funcionamiento de este *framework* mediante su evaluación con métricas de desarrollo estándar en el área. Finalmente, se presentan dos acercamientos de post procesamiento para filtrar y recuperar personas en las detecciones y así mejorar los resultados del sistema.

Marco teórico

El campo de la visión por computadora (*computer vision*) ha evolucionado significativamente en los últimos años (Wu *et al.*, 2020). Gracias a la investigación en vehículos autónomos (Ghari *et al.*, 2022), robótica y los grandes avances en inteligencia artificial, revolucionando incluso los motores de búsqueda (Xu *et al.*, 2019), este campo ha dejado de ser de nicho para convertirse en uno altamente difundido.

Las herramientas de libre acceso han permitido a una amplia variedad de personas incursionar en el campo de *computer vision*. La biblioteca OpenCV (Bradski, 2000) es una demostración de esto, junto con TensorFlow (Abadi *et al.*, 2016) y PyTorch (Paszke *et al.*, 2019), dos reconocidas bibliotecas de aprendizaje automático. Además, ONNX (2017) y ailia MODELS (Ax Inc., 2021) también son herramientas accesibles que han llevado a que entrenar, correr, desarrollar e implementar algoritmos y sistemas de *computer vision* no sea una tarea reservada únicamente a la comunidad científica o académica.

Esto ha generado la necesidad de ofrecer cada vez más algoritmos de fácil implementación y con pocos requisitos computacionales como los modelos YOLO (Redmon y Farhadi, 2018), así como herramientas de evaluación y conjuntos de datos anotados (*dataset*) como Caltech Pedestrian (Dollar *et al.*, 2009), que contiene imágenes y videos de peatones en diversos escenarios; MOTChallenge (Leal-Taixé *et al.*, 2015), que se utiliza como entrenamiento y prueba del desafío MOT y contiene videos de zonas exteriores e interiores con mucho movimiento de personas; CrowdHuman (Shao *et al.*, 2018), enfocado en multitudes cuyo contenido suma más de 2,5 millones de anotaciones de personas; y PersonPath22 (Shuai *et al.*, 2022), un *dataset* muy reciente de gran escala que ofrece variaciones en el ángulo de cámara y calidad de iluminación.

Etiquetado y validación

Para avanzar en la tecnología es necesario evaluar los resultados de los algoritmos. Esto requiere la obtención de métricas de validación para ser comparados. Herramientas como TrackEval (Luiten y Hoffhues, 2020) permiten obtener diversas métricas están- dar en el área, como IDF1 de Identity que evalúa la precisión de la identificación de objetos en una tarea de seguimiento (Ristani *et al.*, 2016), SFDA de VACE que analiza la superposición espacial entre las detecciones del algoritmo y la verdad absoluta (*ground truth*) para ponderar la detección (Manohar, Soundararajan *et al.*, 2006), y métricas de exactitud de seguimiento de orden superior de HOTA (Luiten *et al.*, 2020). Dichas métricas se obtienen a partir de la salida de los algoritmos y la *ground truth* de los *dataset*s, obtenida con los datos anotados o con su etiquetado manual, por ejemplo, con CVAT (Openvinotoolkit, 2018).

Las métricas también se han adaptado al estado actual de la tecnología, las recientes métricas HOTA son un ejemplo de esto, ofreciendo una métrica unificada y directa para evaluar el resultado de un algoritmo. Se obtiene la descomposición de esta en cinco métricas adicionales que abordan cada tipo de error posible en los algoritmos de seguimiento: Precisión de Detección (*detection precision*), Exhaustividad de Detección (*detection recall*), Precisión de Asociación (*association precision*), Exhaustividad de Asociación (*association recall*) y Exactitud de Localización (*localization accuracy*), lo cual permite a desarrolladores e investigadores identificar las componentes que fallan (Manohar, Boonstra *et al.*, 2006). HOTA mide qué tan bien las trayectorias de detecciones coincidentes se alinean, incrementando su efectividad mediante la utilización del algoritmo húngaro (Malkoff, 1997) para seleccionar las coincidencias entre detecciones y *ground truth* (GT). En particular, la precisión de detección mide la cantidad de detecciones correctas en relación a la cantidad de detecciones totales. En cambio, la exhaustividad de detección mide la cantidad de detecciones correctas en proporción a todos los objetos existentes en la *ground truth*.

YOLO

YOLO (*You only look once*) es un algoritmo de detección de objetos que puede utilizarse tanto con imágenes como con videos (Redmon *et al.*, 2015). Su funcionamiento se basa en dividir la imagen a analizar en una cuadrícula de celdas para luego predecir para cada una, un set de cuadros delimitadores junto con la probabilidad de que el objeto detectado pertenezca a cada clase, siendo una clase un tipo de objeto.

Este algoritmo utiliza una sola red neuronal para hacer estas predicciones para toda la imagen en una sola iteración. Su arquitectura consiste en una red neuronal convolucional (CNN) que extrae las características de la imagen (O'Shea y Nash, 2015), seguida por varias capas totalmente conectadas que se encargan de predecir los cuadros delimitadores y el tipo de objeto detectado con su probabilidad.

SiamMOT

La red SiamMOT (*Siamese multi-object tracking*) de seguimiento de objetos en videos (Shuai *et al.*, 2021), utiliza una arquitectura de redes neuronales siamesas y está compuesto por dos ramas idénticas que comparten los mismos pesos. Una rama es responsable de procesar la imagen actual del video y la otra rama procesa una imagen de referencia. La red utiliza un extractor de características basado en CNN para obtener características discriminativas de las imágenes procesadas por cada rama.

Además, SiamMOT cuenta con un módulo de coincidencia basado en FCN (*Fully Convolutional Network*) que compara las características de las dos ramas para determinar la posición del objeto en el siguiente *frame* del video (Long *et al.*, 2014). Este módulo de coincidencia utiliza una técnica llamada "correlación cruzada" que permite medir la similitud entre las características de las dos ramas. Al realizar la etapa de entrenamiento, utiliza un módulo de optimización basado en el gradiente estocástico descendente (SGD) para actualizar los pesos de la red neuronal y mejorar el desempeño del modelo en cada iteración. En esta etapa se utiliza una función de pérdida que mide la discrepancia entre la posición del objeto predicha por el modelo y la posición real del objeto en el siguiente *frame* del video.

Mapa de Calor

Los mapas de calor han sido una herramienta importante para la visualización de datos espaciales y temporales complejos desde hace mucho tiempo (Wilkinson y Friendly, 2009).

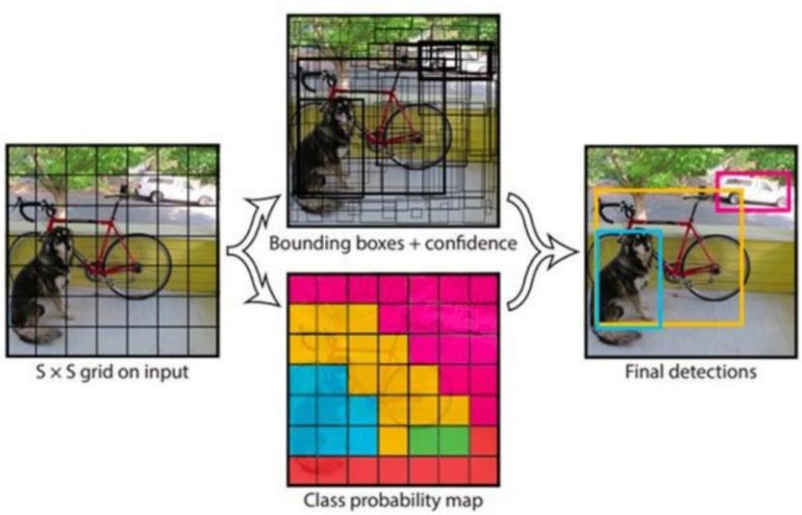


Fig. 1. Funcionamiento de Yolo (imagen reproducida con autorización de Redmon J.)

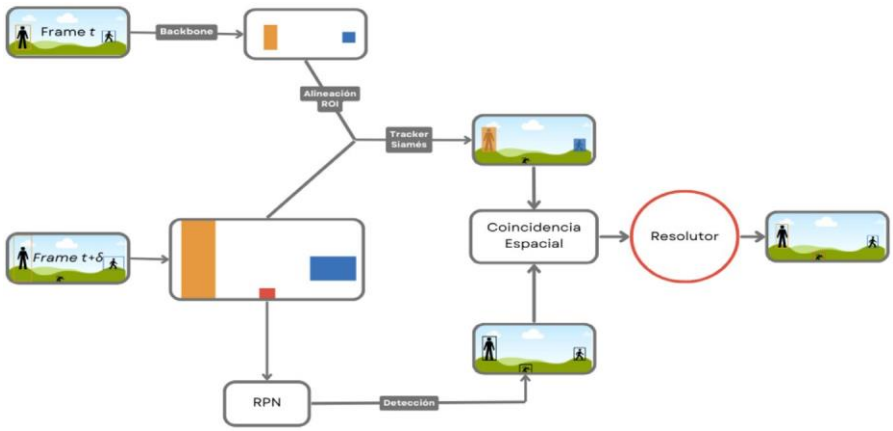


Fig. 2. SiamMOT

Permiten la identificación rápida y eficiente de patrones, tendencias y anomalías en grandes conjuntos de datos. La utilidad de los mapas de calor se ha demostrado en diversas áreas de investigación, como la ciencia ambiental, la epidemiología y la planificación urbana. En la literatura científica se han propuesto diferentes métodos para crear mapas de calor y se ha evaluado su eficacia en la representación de diferentes tipos de datos. En este trabajo se utiliza para identificar las áreas de un comercio con menor y mayor ocupación, proporcionando información valiosa para la comprensión del comportamiento de los clientes y la toma de decisiones empresariales.

Parte Experimental

En la Figura 3 se muestra la secuencia de comandos (*pipeline*) desarrollado para la solución. El primer paso es la adquisición de video, que consiste en un archivo de extensión MP4 con una tasa de fotogramas de 24 FPS, obtenido a través de una cámara de vigilancia. El video es preprocesado realizando una redimensión de la imagen para aumentar la precisión de las detecciones.

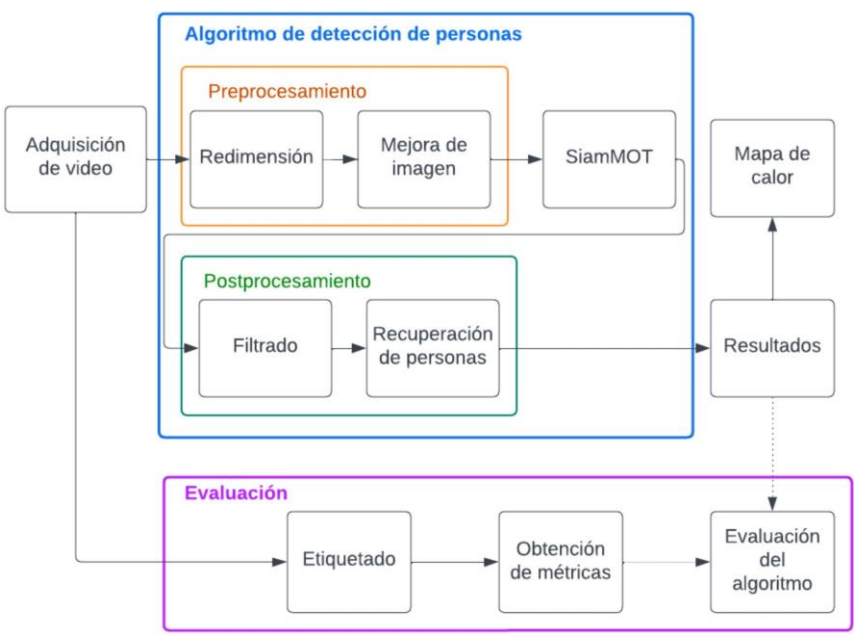


Fig. 3. *Pipeline* de la solución

En este trabajo se utiliza el modelo preentrenado de SiamMOT a través de la biblioteca ailia MODELS para su implementación. SiamMOT fue elegido debido a su precisión y eficacia en la detección de personas. Cabe destacar que, a pesar de que su velocidad de procesamiento es baja en comparación con otros algoritmos, se lo considera una mejor opción debido a su capacidad para detectar de manera efectiva a las personas en una librería. La salida del algoritmo se obtiene en formato MOT1.1 para evaluar la solución propuesta (Leal-Taixé *et al.*, 2015).

El resultado de la detección de personas es enviado a la etapa de postprocesamiento, donde se filtran las detecciones erróneas y se recuperan las personas que no fueron detectadas en el primer análisis. Estos resultados son enviados al generador del mapa de calor para aportar los datos al comercio. Por otro lado, para evaluar el rendimiento de la solución se utilizan métricas de desarrollo estándar, incluyendo las métricas Identity, VACE y HOTA, obtenidas con TrackEval a partir de los videos previamente etiquetados manualmente con CVAT.

Preprocesamiento

Se realizaron pruebas para mejorar la detección de personas en el video, tales como la reducción de FPS y la división de la imagen en cuatro partes iguales, con el objetivo de enfocar el algoritmo en una región más acotada. Sin embargo, estas pruebas presentan algunos desafíos como la dificultad para detectar personas en segundo plano. La reducción de la cantidad de FPS fue suficiente para optimizar los recursos computacionales sin perder precisión en la detección de personas como se presenta en la Tabla 1.

FPS	Tiempo [s]	RAM [MB]	CPU [%]	Precisión	Recall
24	1763	870	60	94.18	80.28
12	943	860	57	96.60	79.28
8	598	900	52	98.16	76.20

Tabla 1. Comparativa entre diferentes FPS

El preprocesamiento incorporado consiste en redimensionar y recortar el video tomando la región de interés (ROI). Para este propósito, se utiliza la biblioteca de OpenCV para *Python* con las dimensiones esperadas por el modelo de seguimiento de personas.

Postprocesamiento

SiamMOT demostró una gran capacidad para reconocer a las personas, sin embargo, se han encontrado casos en los que se producen resultados erróneos. Para abordar este problema, se han utilizado diversas estrategias. En particular, se observó que el modelo produce resultados de tamaño incoherente en algunos casos, por lo que se implementó un filtro para eliminar las detecciones que superen una altura umbral no habitual paralas personas.

Para determinar cuáles detecciones deben ser filtradas, se utilizaron los parámetros de la cámara tales como su posición e inclinación y su distancia focal (Li *et al.*, 2015). Con ellos, y tomando en cuenta la altura de la detección en el *frame*, se estimó la altura real de la detección.

Altura real de la detección:

$$H_d = \frac{-f \cdot H_c \cdot (\theta + 1) \cdot h_d}{tan\theta \cdot y_{d0} \cdot y_{d1} - f \cdot y_{d1} + f \cdot \theta \cdot y_{d0} - f^2 \cdot tan\theta} \qquad (1)$$

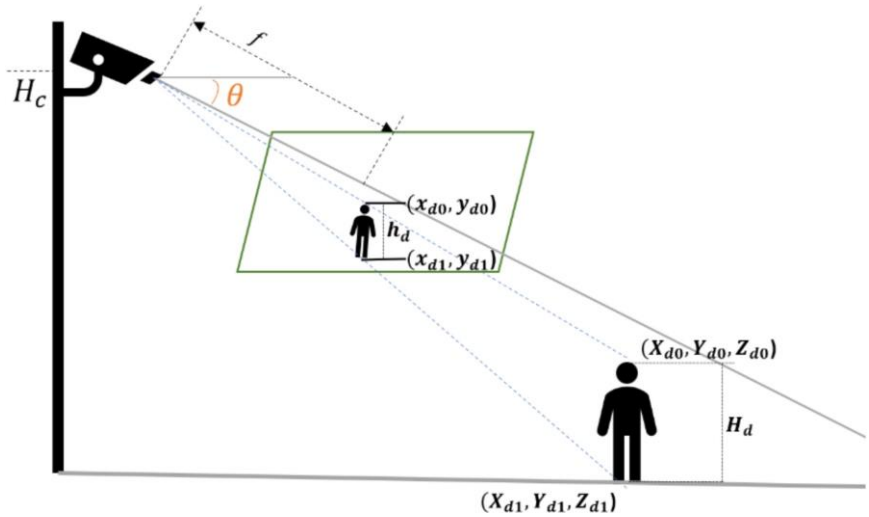


Fig. 4. Detección con cámara de vigilancia

Tabla 2. Símbolos y definiciones

Símbolo	Definición	Unidad
f	Distancia focal	px
Θ	Ángulo de inclinación de la cámara	°
(x_{d0}, y_{d0})	Posición de la cabeza en el *frame*	(px, px)
(x_{d1}, y_{d1})	Posición del pie en el *frame*	(px, px)
h_d	Altura de la detección en el *frame*	px
H_C	Altura de la cámara	m
(X_{d0}, Y_{d0}, Z_{d0})	Posición de la cabeza en el plano real	(m, m, m)
(X_{d1}, Y_{d1}, Z_{d1})	Posición del pie en el plano real	(m, m, m)
H_d	Altura de la detección real	m

Si los valores obtenidos para la altura de una detección son incoherentes, la detección se filtra y se descarta. De esta manera, se asegura que los resultados obtenidos sean precisos y confiables filtrando falsos positivos.

Debe notarse que en las fórmulas presentadas no se tiene en cuenta la distorsión de la cámara, la cual es desconocida y despreciable, por lo que se trata de una estimación. Sólo se ha decidido filtrar detecciones de tamaño mayor que dos metros y no de menor, ya que, al haber oclusiones parciales, sólo es posible saber el tamaño de la parte visible y no del total.

En la Figura 5 se filtra la detección roja en el margen izquierdo por ser errónea dada las dimensiones y la posición en la imagen. Visualmente, puede ser comparada con las personas adyacentes, ya que incluye otras dos detecciones y sería imposible que una persona sea de esa altura.

Fig. 5. Filtrado de detecciones por tamaño

Además, se implementó un filtro para las detecciones que cumplan los siguientes criterios en conjunto: ser detectadas en un *frame* diferente al inicial, haber sido detectadas por menos de 50 *frames*, haber sido inicialmente detectadas lejos del borde de la imagen y tener un promedio de probabilidad de ser persona inferior al 50 %. El objetivo de estas validaciones es eliminar los falsos positivos que aparezcan en la imagen, sin haber ingresado por un borde de esta, asegurando que no influyan negativamente por un período prolongado.Dos ejemplos de detecciones erróneas filtradas se muestran en la Figura 6.

Una vez realizado el filtrado, se propone una etapa de recuperación de personas enfocada en encontrar las detecciones correctas de una persona que desapareció temporalmente debido a oclusiones parciales. La mayoría de las veces, la persona vuelve a ser detectada después de un corto período de tiempo con un nuevo identificador y en una nueva posición. Para abordar este desafío, se propone un método basado en la trayectoria del objeto (Kalman, 1960).

En primer lugar, se analizaron los videos del caso en estudio para estimar que, en promedio, una persona tarda menos de 10 segundos en pasar por un pasillo con otras personas o en buscar un libro de una estantería baja.

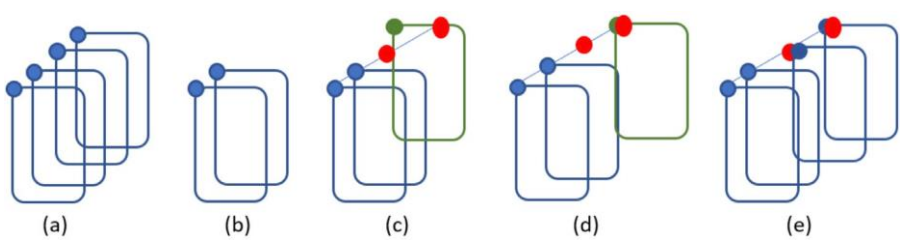
Basándose en esta información, se utilizó la trayectoria de la persona para seguir su movimiento y buscar una nueva detección cuando se deja de detectar.

Dado que la velocidad y la dirección de las personas no son constantes, se estimó un radio de búsqueda que aumenta de acuerdo con los *frames* transcurridos sin encontrar la detección. Si transcurrido un tiempo, una nueva detección aparece en el radio de búsqueda, se asociará a la perdida y se completarán las detecciones interpolando su posición utilizando la velocidad, la trayectoria con la que venía y la posición final encontrada. En caso de que no se encuentre una nueva detección dentro del tiempo estipulado, se descarta la búsqueda.

En la Figura 7 se ilustra el proceso de recuperación de personas. En la imagen (a) se muestra el recorrido de una persona usando su cuadro delimitador *(bounding box)* sin perder

Fig. 6. Filtrado de detecciones

Fig. 7. Recuperación de personas



ninguna detección a lo largo de cuatro *frames*. En el caso que esta persona se oculte, se dejará de detectar (b) perdiendo las últimas dos detecciones. El método busca recuperar estas detecciones perdidas, completando los *frames* con la información de la nueva detección encontrada. El proceso de recuperación comienza siguiendo el recorrido histórico de la detección y busca en los siguientes *frames*, marcados con un círculo rojo. El área roja de búsqueda incrementa proporcionalmente a los *frames* transcurridos. La detección verde en la imagen (c) se encuentra fuera del rango de búsqueda, por lo que no es asociada y sigue buscando en los siguientes *frames*. En la imagen (d) se encuentra una detección en el radio de búsqueda, que subsecuentemente es asociada como la misma detección, reemplazando el nuevo identificador con el anterior. Las detecciones faltantes son creadas utilizando el recorrido y la nueva detección encontrada (e).

Este método puede agregar falsos positivos y disminuir la precisión, pero se utiliza para aumentar significativamente el *recall* en el algoritmo, que tiene originalmente una alta precisión y bajo *recall*.

En la secuencia de la Figura 8 se muestra cómo se deja de detectar temporalmente una persona por una oclusión (b) debido a otra persona, pero vuelve a ser detectada transcurrido un tiempo (c). En la siguiente fila, se utilizó el método de recuperación, con el que se recuperaron todos los *frames* dónde no se había detectado a esta persona.

(a) Frame 279

(b) Frame 283

(c) Frame 493

Fig. 8. Recuperación de personas

Mapa de calor

La creación de mapas de calor permite analizar y visualizar de manera efectiva las zonas más frecuentadas por los clientes en un establecimiento. A partir de esta información, se obtienen indicadores valiosos utilizados por los negocios para mejorar la experiencia del cliente y aumentar las ventas. A continuación, se presenta el desarrollo del mapa de calor para el análisis del comportamiento del cliente en una tienda minorista.

La generación del mapa de calor por sustracción de fondo consta de varios pasos, comenzando con el almacenamiento del primer *frame* y la inicialización de una imagen completamente negra con las mismas dimensiones. En una segunda instancia se recorren todos los *frames* restantes y se realiza la sustracción de fondo, identificándose los píxeles de interés y sumándolos para obtener una imagen saturada. Dicha imagen es una matriz de píxeles que representa el espacio ocupado por las personas detectadas. Se identifica el píxel más intenso en la matriz y se normalizan todos los píxeles para obtener un mapa de calor que no esté saturado. A continuación, los píxeles normalizados se convierten en una imagen en escala de grises. Por último, se utiliza un identificador de contornos con un umbral para indicar la máxima lejanía permitida entre píxeles que conforman una misma figura, y de esa forma, se identifica el área de mayor ocupación del mapa de calor.

Durante el proceso de generación del mapa de calor mediante el uso de *bounding box* se llevan a cabo una serie de pasos. En primer lugar, se almacenan los *bounding box* de las detecciones realizadas y de la *ground truth* correspondiente del video. Posteriormente, se generan máscaras para cada cuadro, las cuales consisten en un sello de todos los *bounding box* sobre una imagen negra por *frame*, contemplando la superposición de varios *bounding box* en un mismo *frame*, ya que esto corresponde a una mayor permanencia en el lugar debido a que dos personas están en la misma zona. A continuación, se aplica la técnica de sumar las matrices de todas las máscaras creadas y tomar el valor del mayor píxel, para así generar

Fig. 9. Mapa de calor por sustracción de fondo

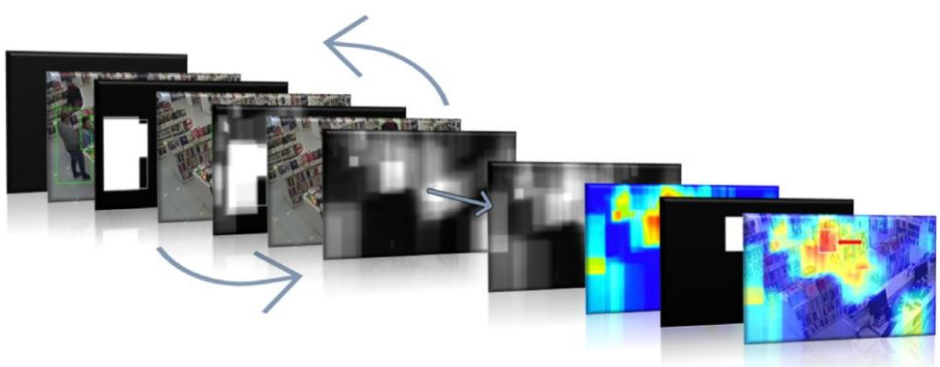


Fig. 10. Mapa de calor por *bounding box*

el mapa de calor en escala de grises y realizar un filtrado de la zona más ocupada ubicando el contorno de los píxeles más intensos. Finalmente, se genera la métrica de desarrollo propia del mapa de calor al aplicar la Intersección sobre la Unión (IoU) del *bounding box* final del lugar más ocupado detectado frente al *bounding box* del lugar más ocupado de la *ground truth*. Esto permite medir la precisión del algoritmo de detección utilizado.

La generación del mapa de calor a partir de las detecciones obtenidas permite incluir en él a ambas personas, la que ocluye y la ocluida, mientras que, en la sustracción de fondo en presencia de oclusiones, se genera el mapa de calor sólo para la persona que se encuentra en el frente mediante la obtención de una figura representativa, lo que la hace menos precisa.

Resultados y Discusión

Con el propósito de analizar la dinámica de la clientela, se empleó la información obtenida mediante la detección de personas en cada *frame* para determinar el instante de tiempo de mayor afluencia (máximas personas detectadas - MPD) en el área bajo observación. A través de la relación existente entre la cantidad de *frames* por segundo y la hora inicial del video, se obtuvo la distribución temporal de la cantidad de personas en la escena. Este enfoque permitió identificar satisfactoriamente los momentos de mayor afluencia de personas y, en consecuencia, contribuyó a un análisis más detallado de la dinámica de la clientela.

$$hora_{frame} = hora_{Inicial} + \frac{frame}{fps \cdot 3600} \quad (2)$$

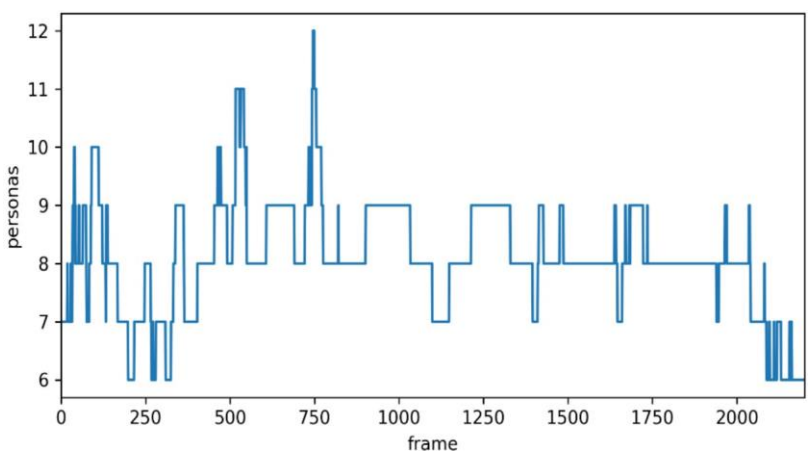


Fig. 11. Personas detectadas por frame

Se complementa la información obtenida del análisis de la cantidad de personas en cada *frame* con un mapa de calor generado a partir de las detecciones del SiamMOT corre- gidas por los filtros previamente explicados. Esta estrategia permite una identificación espacial de las zonas más y menos transitadas, lo que brinda al dueño del establecimientouna mejor comprensión de la ocupación de los clientes dentro del local.

Se aplica la métrica IoU para comparar el *bounding box* obtenido de la zona más ocupada por las detecciones con el *bounding box* obtenido por la *ground truth*, lo que permite validar la información y estimar la precisión del algoritmo.

Fig. 12. Detección 30 minutos

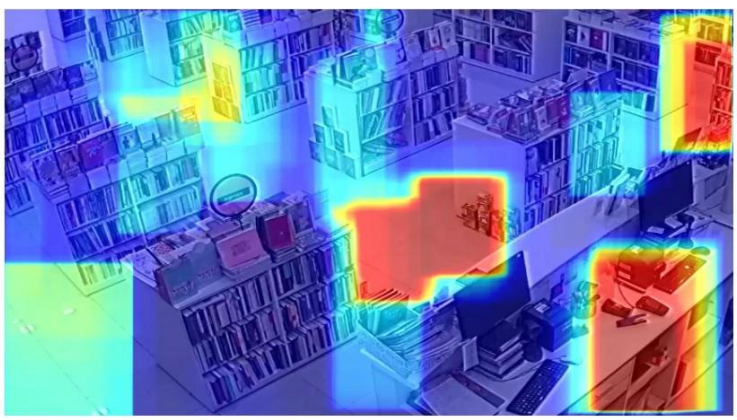


Fig. 13. Detección 90 segundos

Fig. 14. Ground truth 90 segundos

Los principales aspectos de la solución se basaron en la precisión, el *recall* y la confiabilidad de las detecciones. Una vez obtenidas las detecciones correctas (TP) e incorrectas (FP), fue posible determinar las detecciones faltantes (FN) y calcular las siguientes métricas:

Error relativo de cantidad de personas detectadas por *frame* (ERCP):

$$ERCP = \frac{Detectadas - Reales}{(Detectadas, Reales)} \cdot 100\% \qquad (3)$$

Error relativo de cantidad de personas detectadas promedio (ERCPP):

$$ERCPP = \frac{\sum_{1}^{frames} ERCP}{frames} \cdot 100\% \quad (4)$$

Precisión Personas Detectadas (PPD):

$$PPD = \frac{TP}{TP+FP} \quad (5)$$

Recall Personas Detectadas (RPD):

$$RPD = \frac{TP}{TP+FN} \quad (6)$$

F1 Score Personas Detectadas (F1PD):

$$F1PD = \frac{2 \cdot PPD \cdot RPD}{PPD+RPD} \quad (7)$$

Intersección sobre unión (IOUBB):

$$IOUBB = \frac{A \cap B}{A \cup B} \quad (8)$$

Tabla 3. Símbolos y definiciones

Símbolo	Definición	Unidad
$Detectadas$	Personas detectadas en cada *frame*	$personas$
$Reales$	Personas reales en cada *frame*	$personas$
$frames$	Cantidad de *frames* del video	$frames$
TP	Detecciones correctas en cada *frame*	$personas$
FP	Detecciones incorrectas en cada *frame*	$personas$
FN	Personas no detectadas en cada *frame*	$personas$
A	*Bounding box* resultante del mapa de calor de detecciones	(px, px, px, px)
B	*Bounding box* resultante del mapa de calor de la *GT*	(px, px, px, px)

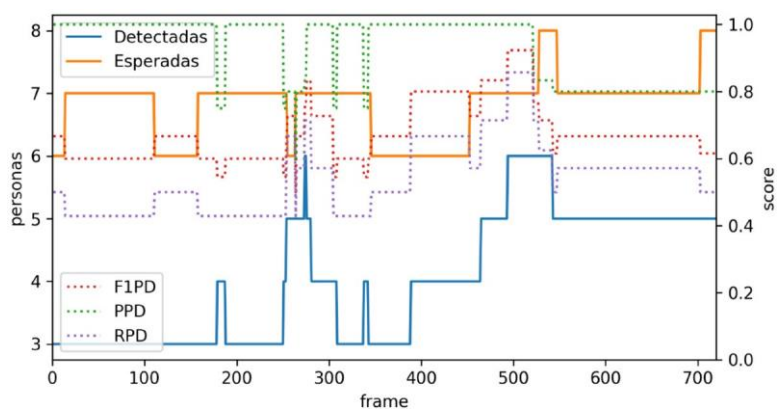


Fig. 15. Métricas de desarrollo propias y de negocio

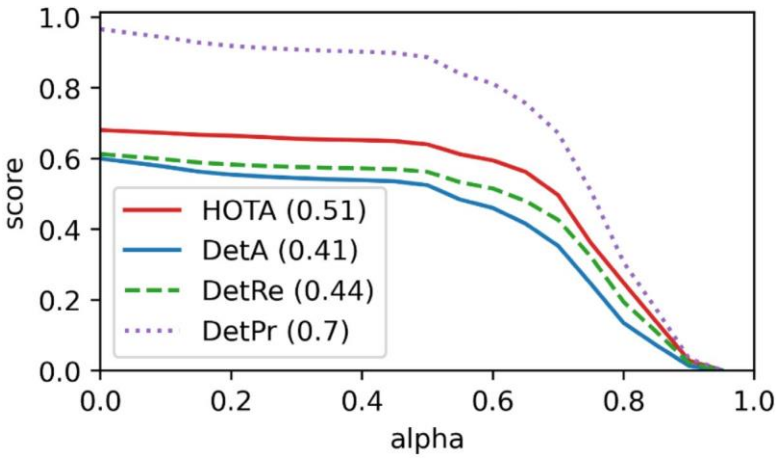


Fig. 16. Métricas de desarrollo estándar

Comparación de métricas

Se presentan en las Tablas 4 y 5 las métricas de desarrollo y negocio obtenidas para un video de 30 segundos, discriminando por algoritmo y técnica de procesamiento. Los resultados reafirman que el algoritmo SiamMOT obtiene métricas superiores en comparación con los demás algoritmos. Además, se observa una clara mejora en las métricas gracias a las técnicas de postprocesamiento aplicadas. Se experimentaron dos variaciones en el postprocesamiento, variando el radio inicial de recuperación de personas entre 5 y 10 píxeles. Si bien se observa una leve variación en las métricas de desarrollo, las métricas de producción no se vieron afectadas por el diámetro elegido, demostrando la robustez del método ante variaciones de parámetros. Sin embargo, se debe tener en cuenta que cuanto mayor sea el radio, más probabilidades habrá de encontrar falsos positivos.

Se observa una mejora significativa gracias al postprocesamiento con un incremento de 6,4% en IDF1 y de 8,2% en el *recall* de detecciones en comparación con la salida del modelo SiamMOT.

La ejecución se realizó en una computadora de uso doméstico con una memoria RAM de 16 GB, un CPU de 8 núcleos con una velocidad de 3,6 GHz. Los recursos consumidospor cada algoritmo se muestran en la Tabla 7, así como el tiempo transcurrido. Para este relevamiento se procesó un video de 43992 *frames*.

Tabla 4. Métricas de desarrollo propias y de negocio de los diferentes modelos

Algoritmo	F1PD	PPD	RPD	MPD	IOUBB [%]
YOLOv3	0,4034	0,9952	0,2608	3/8	0,9034
YOLOv7	0,6206	0,9923	0,4624	5/8	0,9292
YOLOx	0,4815	0,9995	0,3235	4/8	0,9296
SiamMOT	0,6759	0,9295	0,5417	6/8	0,9409
Algoritmo radio 5	**0,7090**	**0,9302**	**0,5866**	6/8	0,9409
Algoritmo radio 10	**0,7090**	**0,9302**	**0,5866**	6/8	0,9409

Tabla 5. Métricas de desarrollo estándar de los diferentes modelos

Algoritmo	HOTA				Identity	VACE	
	HOTA	DetA	DetRe	DetPr	IDF1	SFDA	ATA
YOLOv3	28,350	18,034	18,488	70,817	35,142	28,462	23,534
YOLOv7	25,390	33,482	34,604	74,275	32,054	45,567	17,969
YOLOx	21,246	23,773	24,361	75,121	24,187	35,610	11,748
SiamMOT	49,973	38,042	41,424	69,980	58,356	50,786	34,329
Algoritmo radio 5	**50,757**	**40,593**	**44,338**	69,887	**62,135**	**52,923**	**39,002**
Algoritmo radio 10	**50,840**	**40,767**	**44,473**	70,100	**62,111**	**53,085**	**38,989**

Tabla 6. Definición de métricas

HOTA	Métrica unificada y directa para evaluar el resultado de un algoritmo de seguimiento de objetos en videos. Mide que tan bien se alinean las trayectorias de detecciones coincidentes entre la *GT* y lo detectado por el algoritmo.
DetA	Exactitud de detección, conformada a partir del *recall* y la precisión de detección
DetRe	Detecciones correctas del algoritmo sobre total de detecciones en la *GT* para distintos umbrales de localización promediados.
DetPr	Detecciones correctas del algoritmo respecto al total de detecciones para distintos umbrales de localización promediados.
IDF1	Relación entre las detecciones correctamente identificadas y el número promedio de detecciones tanto de la *GT* como calculadas.
SFDA	Exactitud de la detección para una secuencia de *frames*.
ATA	Precisión del seguimiento para cada objeto individual.

Tabla 7. Consumo de recursos para diferentes modelos

Algoritmo	Tiempo [s]	RAM[MB]	CPU [%]
YOLOv3	2088,30	126,1	16,1
YOLOv7	2815,89	150,6	13,3
YOLOx	1455,52	200,4	15,0
SiamMOT	28448,94	802,0	60,0

Conclusiones

En este estudio se ha presentado una solución efectiva para el análisis del comportamiento de clientes mediante detección de personas en presencia de oclusión. Se evaluaron varios algoritmos de detección de objetos, de los cuales se seleccionaron el SiamMOT por su gran desempeño en comparación con otros algoritmos investigados y el YOLOv7 por su rápida velocidad de procesamiento.

Además, se aplicaron técnicas de preprocesamiento y postprocesamiento para mejorar la precisión y el *recall* del modelo, incluyendo la utilización de un algoritmo de regresión para estimar la ubicación de la persona durante la oclusión. Estas técnicas demostraron ser efectivas para mejorar la salida del algoritmo, incrementando la SFDA en un 4,52 % y el *recall* en un 7,36 %, lo que también conlleva una mejora de la HOTA en un 1,73 %.

Los resultados obtenidos demuestran que la solución propuesta con ambos algoritmos es satisfactoria. Si bien es cierto que SiamMOT detecta mejor, no se puede ignorar el hecho de que, en esta etapa de la tecnología, YOLOv7 tarda solo el 10% del tiempo y utiliza el 25 % de los recursos, obteniendo resultados aceptables.

Se proporcionaron métricas valiosas para el estudio y evaluación del comportamiento de los clientes en el negocio, lo que permitió identificar las áreas más ocupadas mediante el uso del mapa de calor y los momentos más concurridos mediante las detecciones por *frame*. La generación del mapa de calor por medio de sustracción de fondo no contempla la oclusión, lo cual podría abordarse en un futuro y ser aún más preciso en los resultados. Sin embargo, a medida que el video es más largo, tiene la capacidad de obtener resultados similares al realizado por medio de las detecciones, que contemplan este problema.

Agradecimientos

Agradecemos al Grupo de Inteligencia Artificial y Robótica (GIAR), a los profesores Mg. Ing. Mariano Vidal, Ing. Fernando Valenzuela y Mg. Ing. Pablo Sánchez por su apoyo y guía. A Julián Morrone y Agustín Saco por su gestión para la obtención de datos y recursos de la librería, y a las herramientas de acceso libre utilizadas, tales como OpenCV, ailia MODELS, OnnX, que permiten que el estado del arte llegue a mano de todos fácilmente.

Referencias

ABADI, M.; BARHAM, P.; CHEN, J.; CHEN, Z.; DAVIS, A.; DEAN, J.; DEVIN, M.; GHEMAWAT, S.; IRVING, G.; ISARD, M.; KUDLUR, M.; LEVENBERG, J.; MONGA, R.; MOORE, S.; MURRAY, D. G.; STEINER, B.; TUCKER, P.; VASUDEVAN, V.; WARDEN, P.; … ZHENG, X., (2016). TensorFlow: A system for large-scale machine learning. http://arxiv.org/abs/1605.08695

AHMED, M.; HASHMI, K. A.; PAGANI, A.; LIWICKI, M.; STRICKER, D. y AFZAL, M. Z., (2021). Survey and Performance Analysis of Object Detection in Challenging Environments. https://doi.org/10.20944/preprints202106.0590.v1

Ax Inc., (2021). ailia MODELS. Ax Inc. https://github.com/axinc-ai/ailia-models

BRADSKI, G., (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools. https://www.drdobbs.com/open-source/the-opencv-library/184404319

CHAVA, S.; OETTL, A.; SINGH, M. y ZENG, L., (2022). Creative Destruction? Impact of E-Commerce on the Retail Sector. https://doi.org/10.3386/w30077

DOLLAR, P.; WOJEK, C.; SCHIELE, B. y PERONA, P., (2009). Pedestrian detection: A benchmark. 2009 IEEE Conference on Computer Vision and Pattern Recognition, 304-311. https://doi.org/10.1109/CVPR.2009.5206631

GHARI, B.; TOURANI, A. y SHAHBAHRAMI, A., (2022). A Robust Pedestrian Detection Approach for Autonomous Vehicles. 2022 8th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS), 1-5. https://doi.org/10.1109/ICSPIS56952.2022.10043934

KALMAN, R. E., (1960). A New Approach to Linear Filtering and Prediction Problems. Journal of Basic Engineering, 82 (1), 35-45. https://doi.org/10.1115/1.3662552

LEAL-TAIXÉ, L.; MILAN, A.; REID, I.; ROTH, S. y SCHINDLER, K., (2015). MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking. http://arxiv.org/abs/1504.01942

LI, S.; NGUYEN, V. H.; MA, M.; JIN, C.-B.; DO, T. D. y KIM, H., (2015). A simplified nonlinear regression method for human height estimation in video surveillan-ce. EURASIP Journal on Image and Video Processing, 2015 (1), 32. https://doi.org/10.1186/s13640-015-0086-1

LONG, J.; SHELHAMER, E. y DARRELL, T., (2014). Fully Convolutional Networks for Semantic Segmentation. http://arxiv.org/abs/1411.4038

LUITEN, J. y HOFFHUES, A., (2020). TrackEval. https://github.com/JonathonLuiten/TrackEval.

LUITEN, J.; OSEP, A.; DENDORFER, P.; TORR, P.; GEIGER, A.; LEAL-TAIXE, L. y LEIBE, B., (2020). HOTA: A Higher Order Metric for Evaluating Multi-Object Tracking. International Journal of Computer Vision, 129 (2), 548-578. https://doi.org/10.1007/s11263-020-01375-2

MALKOFF, D. B., (1997). Evaluation of the Jonker-Volgenant-Castanon (JVC) assignment algorithm for track association (I. Kadar, Ed.; p. 228). https://doi.org/10.1117/12.280801

MANE, P., (2022). Global Cloud Services Market 2022 Growing Adoption | Anticipated to Grow $1,429,672.6 Mn in 2030 at a CAGR of 12.3%. Altus Market Research.

MANOHAR, V.; BOONSTRA, M.; KORZHOVA, V.; SOUNDARARAJAN, P.; GOLDGOF, D.; KASTURI, R.; PRASAD, S. y BOWERS, R., (2006). PETS vs VACE Evaluation Programs: A Comparative Study. PETS vs. VACE Evaluation Programs: A Comparative Study.

MANOHAR, V.; SOUNDARARAJAN, P.; RAJU, H.; GOLDGOF, D.; KASTURI, R. y GAROFOLO, J., (2006). Performance Evaluation of Object Detection and Tracking in Video (pp. 151-161). Springer Berlin Heidelberg. https://doi.org/10.1007/11612704_16

NING, C.; MENGLU, L.; HAO, Y.; XUEPING, S. y YUNHONG, L., (2021). Survey of pedestrian detection with occlusion. Complex & Intelligent Systems, 7 (1), 577-587. https://doi.org/10.1007/s40747-020-00206-8

ONNX. (2017). https://github.com/onnx/onnx.

OPENVINOTOOLKIt, O. T., (2018). Powerful and efficient Computer Vision Annota-

tion Tool (CVAT). En GitHub. https://github.com/openvinotoolkit/cvat

O'SHEA, K. y NASH, R., (2015). An Introduction to Convolutional Neural Networks. http://arxiv.org/abs/1511.08458

PASZKE, A.; GROSS, S.; MASSA, F.; LERER, A.; BRADBURY, J.; CHANAN, G.; KILLEEN, T.; LIN, Z.; GIMELSHEIN, N.; ANTIGA, L.; DESMAISON, A.; KÖPF, A.; YANG, E.; DEVITO, Z.; RAISON, M.; TEJANI, A.; CHILAMKURTHY, S.; STEINER, B.; FANG, L.; … CHINTALA, S., (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. http://arxiv.org/abs/1912.01703

REDMON, J.; DIVVALA, S.; GIRSHICK, R. y FARHADI, A., (2015). You Only Look Once: Unified, Real-Time Object Detection. http://arxiv.org/abs/1506.02640

REDMON, J. y FARHADI, A., (2018). YOLOv3: An Incremental Improvement. http://arxiv.org/abs/1804.02767

RISTANI, E.; SOLERA, F.; ZOU, R. S.; CUCCHIARA, R. y TOMASI, C., (2016). Performance Measures and a Data Set for Multi-Target, Multi-Camera Tracking. http://arxiv.org/abs/1609.01775

SALMON, M. y PARMAR, A., (2022). Cloud Computing at Unitec (March 2022). https://doi.org/10.13140/RG.2.2.16483.63528

SHAO, S.; ZHAO, Z.; LI, B.; XIAO, T.; YU, G.; ZHANG, X. y SUN, J., (2018). CrowdHuman: A Benchmark for Detecting Human in a Crowd. http://arxiv.org/abs/1805.00123

SHUAI, B.; BERGAMO, A.; BUECHLER, U.; BERNESHAWI, A.; BODEN, A. y TIGHE, J., (2022). Large Scale Real-World Multi-Person Tracking. http://arxiv.org/abs/2211.02175

SHUAI, B.; BERNESHAWI, A.; LI, X.; MODOLO, D. y TIGHE, J., (2021). SiamMOT: Siamese Multi-Object Tracking. https://www.amazon.science/publications/siammot-siamese-multi-object-tracking

WILKINSON, L. y FRIENDLY, M., (2009). The History of the Cluster Heat Map. The American Statistician, 63 (2), 179-184. https://doi.org/10.1198/tas.2009.0033

WU, X.; SAHOO, D. y HOI, S. C. H., (2020). Recent advances in deep learning for object detection. Neurocomputing, 396, 39-64. https://doi.org/10.1016/j.neucom.2020.01.085

XU, J.; HE, X. y LI, H., (2019). Deep Learning for Matching in Search and Recommendation. Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, 832-833. https://doi.org/10.1145/3289600.3291380