



Universidad Católica
San Pablo

**Seguimiento de múltiples objetos basado en un
modelo de Aprendizaje Profundo**

Andre Luí Ramos Provincia

Orientador: Dr. Rensso Victor Hugo Mora Colque

Tesis profesional presentada al Programa Profesional de Ciencia de la Computación como parte de los requisitos para obtener el Título Profesional de Lic. en Ciencia de la Computación.

**UCSP- Universidad Católica San Pablo
Julio de 2020**

Dedicatoria

UNIVERSIDAD CATÓLICA SAN PABLO
FACULTAD DE INGENIERÍA Y COMPUTACIÓN
PROGRAMA PROFESIONAL DE CIENCIA DE LA
COMPUTACIÓN

Seguimiento de múltiples objetos basado en un modelo
de Aprendizaje Profundo

Arequipa, 27 de julio de 2020

Aprobado por:

Prof. Alex Jesús Cuadros Vargas

Prof. José Eduardo Ochoa Luna

Prof. Edward Jorge Yuri Cayllahua Cahuina

Abreviaturas

ACF *Aggregate Channel Features*

CNN *Convolutional Neural Network*

DA *Data Association*

DHN *Deep Hungarian Network*

DPM *Deformable Part-based Model*

HOMHT *Hypothesis-Oriented MHT*

JPDAF *Joint Probabilistic Data Association Filter*

JPDA *Joint Probabilistic Data Association*

MHT *Multiple Hypothesis Tracking*

MOT *Multiple Object Tracking*

MOTA *Multiple Object Tracking Accuracy*

MOTP *Multiple Object Tracking Precision*

PDA *Probabilistic Data Association*

RPN *Region Proposal Network*

SOT *Single Object Tracking*

SNF *Soft-rejection based Network Fusion*

SSD *Single Shot MultiBox Detector*

SVM *Support Vector Machine*

TOMHT *TrackOriented MHT*

YOLO *You Only Look Once*

Agradecimientos

Resumen

El seguimiento de múltiples objetos en entornos reales es un problema desafiante, principalmente porque la silueta deformable del objeto y la iluminación variable del entorno cambian la apariencia de los objetos con el tiempo. Esto causa una alta dificultad en la asociación temporal de la identidad de los objetos. El problema se acentúa cuando los objetos se mueven cerca de otros, se ocluyen o cambian abruptamente su trayectoria. En esta tesis se propone un nuevo modelo para mejorar la precisión y velocidad del seguimiento de objetos en una secuencia de video en tiempo real.

Abstract

Multiple tracking objects in real environments is a challenging problem, mainly because the deformable silhouette of the object and the variable lighting of the environment change the appearance of objects over time. This causes a high difficulty in the temporal association of the identity of objects. The problem is accentuated when objects move close to others, become occluded, or abruptly change their trajectory. In this thesis I propose a new model to improve tracking accuracy.

Índice general

1. Introducción	2
1.1. Motivación y Contexto	3
1.2. Planteamiento del Problema	3
1.3. Objetivos	4
1.3.1. Objetivos Específicos	4
1.4. Organización de la tesis	4
2. Marco Teórico	5
2.1. Multiple Object Tracking	5
2.1.1. Redes Neuronales Convolucionales	6
2.1.2. <i>Region Proposal Network (Region Proposal Network (RPN))</i>	7
2.1.3. Faster R-CNN	7
2.1.4. YOLO	8
2.1.5. Asociación de datos	9
2.1.6. <i>Intersect over Union (IoU)</i>	10
2.1.7. Filtro de Kalman	10
2.1.8. Método Húngaro	10
3. Estado del Arte	11
3.1. Detección de objetos	11
3.2. Seguimiento de Objetos	12

4. Propuesta	14
4.1. Detección de Objetos	15
4.2. Seguimiento de Objetos	17
4.2.1. Problema de asignación	17
5. Pruebas y resultados	19
5.1. Conjuntos de datos	19
5.2. Resultados	19
5.2.1. Evaluación del Desempeño	20
5.2.2. Tiempo de Ejecución	22
6. Conclusiones y Trabajos Futuros	23
6.1. Problemas encontrados	23
6.2. Trabajos futuros o recomendaciones	24
Bibliografía	28

Índice de tablas

4.1. Sistemas en tiempo real sobre PASCAL VOC 2007. Comparación de velocidades y rendimientos para los modelos entrenados con los conjuntos de datos PASCAL VOC 2007 y 2012. Los resultados publicados corresponden a las implementaciones de [J. Redmon et al. 2016].	17
5.1. Métricas del <i>MOT16</i> y <i>MOT17</i> [Luo et al., 2017].	19
5.2. Características de los 2 conjuntos de datos utilizados en las pruebas.	20
5.3. Comparación del rendimiento del <i>Tracking</i> mediante el intercambio de las propuestas y el modelo base del. Estos resultados son evaluados con las secuencias de validación del <i>MOT20</i>	20
5.4. Resultados cuantitativos de las líneas de base en <i>MOT16</i> con datos de prueba [Luo et al., 2017].	20
5.5. Resultados cuantitativos de las líneas de base en <i>MOT17</i> con datos de prueba [Luo et al., 2017].	21
5.6. Resultados cuantitativos de las líneas de base en <i>MOT20</i> con datos de prueba [Luo et al., 2017].	22

Índice de figuras

2.1. Una ilustración de la arquitectura de una CNN, que muestra explícitamente la delimitación de responsabilidades entre las dos GPUs. Una GPU ejecuta las partes de capa en la parte superior de la figura mientras que la otra ejecuta las partes de capa en la parte inferior. Las GPUs se comunican sólo en ciertas capas. La entrada de la red es de 150.528-dimensional, y el número de neuronas en las capas restantes de la red está dado por 253.440-186.624-64.896-64.896-64.896-43. Fuente: [Aloysius and Geetha, 2017]	7
2.2. Detección con YOLO: Se divide la imagen en celdas, típicamente su cuadrícula de 19×19. Cada celda será responsable de predecir 5 casillas delimitadoras (en caso de que haya más de un objeto en esta celda). Esto nos da 1805 bounding boxes con las variables: pc para el objeto predicho , bx y by para el centro de los bounding boxes, bh para la altura, bw para el ancho, para una imagen. Fuente: [Redmon et al., 2016]	8
2.3. Dos conjuntos de ejemplos (a) y (b) con los recuadros delimitadores representados por (a) dos esquinas (x_1, y_1, x_2, y_2) y (b) centro y tamaño (xc, yc, w, h). Para los tres casos en cada conjunto (a) distancia normal L2, $\ \cdot\ _2$, y (b) distancia normal L1, $\ \cdot\ _1$, entre la representación de dos rectángulos son exactamente el mismo valor, pero sus valores <i>IoU</i> y <i>GIoU</i> son muy diferentes. Fuente: [Bochinski et al., 2017]	9
4.1. Pipeline. El canal completo de percepción visual en la conducción autónoma, es decir, la detección de objetos y Seguimiento de objetos múltiples (MOT).	14
4.2. Selección de candidatos en base a puntuaciones unificadas. Los candidatos de la detección y las pistas se visualizan como rectángulos sólidos azules y rectángulos punteados rojos, respectivamente. La detección y las pistas pueden complementarse entre sí para la asociación de datos. Fuente: [Xu et al., 2019].	15
4.3. Arquitectura YOLOv3: se compone de 24 capas convolucionales y 2 capas totalmente conectadas. Fuente: [J. Redmon et al., 2016].	16
5.1. Buenos resultados cualitativos	21

5.2.	Mal resultado cualitativo	21
5.3.	Secuencia del <i>tracking</i> multi-objeto, usando la base de datos UCSP	21
5.4.	Técnica DeepSORT	22
5.5.	Técnica DeepSORTYO	22
5.6.	Secuencia del <i>tracking</i> multi-objeto, usando la base de datos <i>MOT17-04</i> . .	22

Capítulo 1

Introducción

El seguimiento multi-objeto se encarga de estimar la ubicación de objetos en movimiento mediante una secuencia de imágenes o un video. Hoy en día el monitoreo de la trayectoria de personas, autos, etc., son detectados por sistemas inteligentes que proporcionan el análisis de los cambios de posición u otras características. Actualmente se cuenta con una amplia gama de investigaciones sobre el seguimiento multi-objeto. Estos estudios analizan desafíos que se presentan en el proceso del seguimiento, tales como: rotación, deformación, perspectiva de la cámara, cambios de luz y occlusiones. En el pasado reciente, gracias a los avances de las redes neuronales, se ha logrado un gran progreso para el seguimiento de objetos [Zhu et al., 2018], que facilitan el estudio de dichos desafíos.

El seguimiento de objetos es uno de los principales desafíos científicos de la visión por computador que es analizado en dos diferentes ámbitos: *Single Object Tracking* (SOT) [He et al., 2017] tiene como objetivo aprender modelos discriminatorios para seguir un objeto y separarlo del fondo. *Multiple Object Tracking* (MOT) [Luo et al., 2017] estima la trayectoria de varios objetos de acuerdo a las detecciones hechas secuencialmente. Ambas técnicas presentan el mismo proceso de seguimiento de objetos, el cual se divide en dos etapas: (a) La detección clasifica y determina la posición de un objeto en la imagen. Este paso debe realizarse antes del seguimiento debido a la necesidad de encontrar la posición del objeto en cada *frame*. Se debe resaltar que el uso del aprendizaje profundo ha desarrollado nuevos métodos de detección [Fan et al., 2019]. (b) El seguimiento se encarga de fijar un objeto en movimiento y determinar si este es el mismo en el marco anterior. *Data Association* (DA) [Wang et al., 2018] es una de las técnicas que realiza el proceso de asociar pistas con objetos reales. Este paso es una tarea difícil por sí sola, debido a las occlusiones e interacciones de objetivos faltantes en entornos saturados.

MOTChallenge [Milan et al., 2016] creó un *framework* para la evaluación justa de algoritmos de seguimiento de múltiples personas. Esta herramienta tiene como objetivo evaluar una variedad de modelos existentes, cuyos resultados son mostrados en las tablas de comparación del MOT. Se escogió las métricas necesarias para el aporte a la investigación [Chau et al., 2013]. El concepto de MOT viene ligado a dos escenarios: MOT en línea y MOT fuera de línea. Para la investigación es conveniente el uso de MOT en línea que permite hacer rastreos en tiempo real, dado que los objetos de ruta se estiman usando solo las detecciones de cuadros actuales y anteriores. Es decir, ejecuta un seguimiento

multi-objeto en tiempo real.

La tesis propuesta ofrece una modelo modificado, donde los experimentos muestran resultados aceptables en condiciones complejas, tales como: visibilidad parcial de las personas para entrar o salir de la escena y occlusiones persistentes entre las personas. Las pruebas se basaron de acuerdo a las métricas escogidas, que son medidas con el *devkit* que ofrece el *MOTChallenge*. Los resultados son alcanzados gracias al uso de sistemas inteligentes [Milan et al., 2016] para el seguimiento de objetos.

1.1. Motivación y Contexto

El seguimiento de objetos es un tema de investigación muy activo en el área de visión por computadora, lo que conlleva al análisis de la posición y/o trayectoria de objetos, que permite determinar el estado de la trayectoria del objeto: esperando algo, invadiendo un área no permitida, o en caso de peatones al desarrollar una actividad sospechosa, esto relacionando la información de las trayectorias de dos o más individuos. El área de seguimiento de video es actualmente de gran interés debido a sus implicaciones en video vigilancia, seguridad, equipos médicos, sistemas robóticos [Granström and Baum, 2016]. Algunos aeropuertos utilizan un sistema de seguimiento de personas y análisis. Se resalta un *software* de seguimiento de personas basado en video de CrowdVision [Lu et al., 2019], que muestra el comportamiento de toda una población de pasajeros a través de seguimientos de individuos en tiempo real. Esto permite a los aeropuertos actuar de forma decisiva para aumentar la eficiencia y la rentabilidad, mejorando al mismo tiempo la experiencia de sus pasajeros.

Entonces se resalta que el seguimiento de objetos ofrece un contexto para la extracción de información significativa como el movimiento de la escena, sustracción de fondo, clasificación de objetos, interacción del objeto con el fondo y otros objetos de una escena, la identificación humana, el comportamiento de los humanos con objeto y fondo, etc. Y añadiendo los desafíos existentes en el MOT en línea, la creación de sistemas para el seguimiento de objetos va en aumento [Reddy et al., 2015], lo que conlleva a un problema abierto que abre posibilidades de investigación en relación al seguimiento de objetos.

1.2. Planteamiento del Problema

El problema del seguimiento de múltiples personas en un escenario estacionario no controlado, conlleva a estimar la ubicación de cada persona en cada secuencia, y a determinar su trayectoria desde que esta entra hasta que sale de la escena. El seguimiento puede presentar cambios de apariencia de la persona durante la secuencia, obteniendo falsos positivos y falsos negativos en la detección de las personas, y presentar occlusiones parciales o totales por otros objetos fijos o en movimiento. Algunas investigaciones recientes utilizan algoritmos de aprendizaje profundo pesados [Shi et al., 2015], que si bien es inevitable la necesidad de usar estos algoritmos para una mejor precisión, esto conlleva muchas veces a disminuir la velocidad de actualización de rastreo. Los algoritmos utilizados en estas

investigaciones puedes ser modificados utilizando nuevas métricas o añadiendo otras heurísticas que conlleven a la reducción de uso de recursos, permitiendo una mejor velocidad de actualización y precisión del seguimiento.

1.3. Objetivos

En esta tesis se plantea diseñar un modelo para MOT, capaz de ubicar y mantener el tramo correcto de múltiples objetos que pueden ser ocluidos, parcial o totalmente a partir de una secuencia de imágenes con escenarios no controlados adquirida de la base de datos *MOTChallenge*. Se reconstruye el modelo de [Xu et al., 2019] en el proceso de detección y asociación de datos. El uso del modelo base conlleva un alto consumo de recursos computacionales debido a su modelo detección [Bertinetto et al., 2016]. Para un seguimiento en tiempo real no es conveniente usar estos tipos de modelos y si es necesario de debe implementar una nueva forma de asociar los datos que ayude a mejorar la precisión del seguimiento.

1.3.1. Objetivos Específicos

Partiendo del objetivo general se plantean los siguientes objetivos específicos.

- El reemplazo del modelo de detección, comparando con otros modelos que ofrezcan una buena precisión y velocidad. Estos modelos no deben de dar altos niveles de procesamiento.
- Realizar una métrica mejorada al modelo base para la asociación de datos en el proceso de seguimiento de objetos.
- Evaluar y validar los resultados obtenidos con la métricas específicas dadas por el MOT *Benchmark*.

1.4. Organización de la tesis

El presente documento está organizado de la siguiente manera: En el Capítulo 2 desarrollamos el Marco Teórico donde hablamos de: las técnicas, definiciones y algoritmos que utilizamos para desarrollar nuestro objetivo. En el Capítulo 3 encontramos Trabajos Relacionados donde se analiza los trabajos previos que han sido desarrollados para el MOT, enfatizando las ventajas o limitaciones que presentan para ser aplicados en situaciones reales. En el Capítulo 4 se explica el modelo propuesto para el seguimiento de objetos y el algoritmo de correspondencia que permiten mantener el rastro de los objetos. En el Capítulo 5 mostraremos las Pruebas y Resultados obtenidos de esta tesis. Por último, en el Capítulo 6 se enfoca en las conclusiones al relacionar los resultados obtenidos del capítulo anterior.

Capítulo 2

Marco Teórico

2.1. Multiple Object Tracking

La tarea del MOT se divide en la detección de múltiples objetos, el mantenimiento de sus identidades y el rendimiento de sus trayectorias individuales con un video de entrada. Los peatones son los objetos no rígidos típicos. El seguimiento de múltiples objetos se basa en tareas de alto nivel tales como la estimación de poses, el reconocimiento de acciones y el análisis de comportamiento. Por otro lado con el seguimiento de objetos individuales *Single Object Tracking (SOT)* [Fiaz et al., 2018], se centra principalmente en el diseño de modelos de apariencia sofisticados y/o modelos de movimiento para hacer frente a factores complejos como los cambios de escala, las rotaciones fuera del plano y las variaciones de iluminación, el seguimiento de objetos múltiples requiere además la resolución de dos tareas: determinar el número de objetos, que normalmente varía con el tiempo, y mantener sus identidades.

Como punto clave se tiene que enfatizar en cada investigación, los problemas base de seguimiento de objetos tales como: 1) occlusiones frecuentes, 2) inicialización y terminación de pistas, 3) apariencia similar, y 4) interacciones entre múltiples objetos. El objetivo del MOT es encontrar los estados secuenciales óptimos de todos los objetos, que generalmente se pueden modelar realizando una estimación a partir de la distribución condicional de los estados secuenciales.

El MOT se categoriza en: cómo se inicializa la tarea, cómo se procesa y qué tipo de resultado se obtiene. El modo de procesamiento se enfoca en seguimiento en línea y seguimiento fuera de línea. La diferencia es, si se utilizan o no observaciones de futuros marcos cuando se maneja el marco actual. Los métodos de seguimiento en línea, sólo se basan en la información pasada disponible hasta el marco actual, mientras que los enfoques de seguimiento fuera de línea emplean observaciones tanto en el pasado como en el futuro. El tipo de salida puede obtener resultados deterministas y probabilísticos, dependiendo de la aleatoriedad de la salida. El resultado del seguimiento determinista es constante cuando se ejecutan los métodos varias veces, mientras que los resultados son diferentes en los diferentes ensayos en curso de los métodos de seguimiento probabilístico.

Cuando se desarrollan los enfoques MOT, se debe considerar dos temas principales.

El primero mide la similitud entre los objetos por cada *frame* y el segundo recupera la información de identidad basada en la medición de la similitud entre los objetos. En términos generales, la primera cuestión tiene que ver con el modelado de: la apariencia, el movimiento, la interacción, la exclusión y la oclusión. La segunda cuestión aborda el problema de la inferencia.

La apariencia es una clave importante para el cálculo de afinidad en MOT. Sin embargo, a diferencia del seguimiento de un solo objeto, que se centra principalmente en la construcción de un modelo de apariencia sofisticado para discriminar el objeto del fondo, la mayoría de los métodos MOT no consideran el modelado de apariencia como el componente central, aunque puede ser un componente importante. La interacción, también conocido como modelo de movimiento mutuo, captura la influencia de un objeto en otros objetos. En el escenario de la multitud, un objeto experimentaría alguna fuerza de otros agentes y objetos. Por ejemplo, cuando un peatón camina por la calle, ajusta su velocidad, dirección y destino, para evitar colisiones con otros. Dadas las múltiples respuestas de detección y las múltiples hipótesis de trayectoria, generalmente hay dos restricciones. La primera es la llamada exclusión de nivel de detección, es decir, dos respuestas de detección diferentes en la misma trama no pueden asignarse al mismo objetivo. La segunda es la llamada exclusión a nivel de trayectoria, es decir, dos trayectorias no pueden estar infinitamente cerca una de otra.

2.1.1. Redes Neuronales Convolucionales

El uso de *conv-net* aportan mucho al momento de procesar imágenes, y pueden aprender relaciones entrada-salida, donde la entrada una imagen y la salida puede votar un resultado de clasificación. Las *conv-net* están basadas en operaciones de convolución y realizan tareas comunes: (i) detección/categorización de objetos, (ii) clasificación de escenas y (iii) clasificación de imágenes.

La Figura.2.1 muestra la estructura de una *conv-net* compuesta por la capa de convolución. La operación de convolución normalmente recibe una imagen y se hace un filtro que nos da un mapa características, esto ayuda a reducir el tamaño, esta es la capa de *pooling* que se genera después de la capa convolucional. El objetivo de esta red radica en la reducción de las dimensiones espaciales del volumen de entrada para la siguiente capa convolucional, y finalmente se ejecuta la capa clasificadora totalmente conectada. En esta última capa se tendrá tantas neuronas como la cantidad número de clases que se escogió para la predicción.

MOT con *Convolutional Neural Network* (CNN) (redes neuronales convolucionales) consta de múltiples rastreadores de objetos basados en CNN, donde las capas convolucionales compartidas se fijan y se usan para extraer la representación de apariencia, mientras que las capas *Full Connected* específicas del objetivo se actualizan en línea para distinguir el objetivo del fondo. Cuando un objetivo experimenta un error de desviación debido a la oclusión, CNN realiza una actualización para borrar toda la memoria anterior del objetivo.

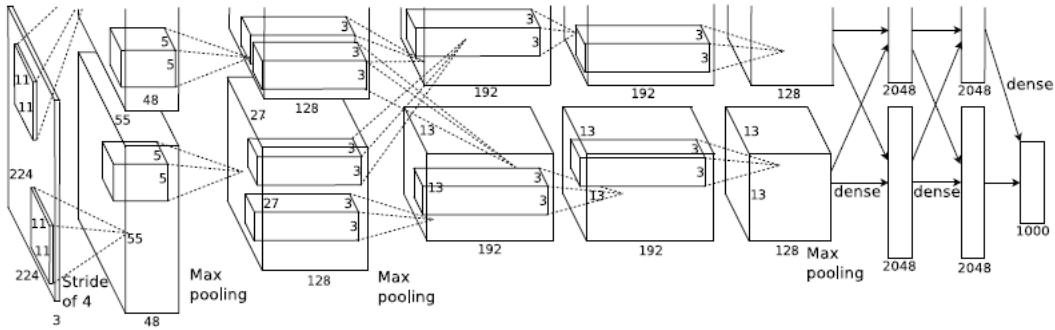


Figura 2.1: Una ilustración de la arquitectura de una CNN, que muestra explícitamente la delimitación de responsabilidades entre las dos GPUs. Una GPU ejecuta las partes de capa en la parte superior de la figura mientras que la otra ejecuta las partes de capa en la parte inferior. Las GPUs se comunican sólo en ciertas capas. La entrada de la red es de 150.528-dimensional, y el número de neuronas en las capas restantes de la red está dado por 253.440-186.624-64.896-64.896-43. Fuente: [Aloysius and Geetha, 2017].

2.1.2. *Region Proposal Network (RPN)*

RPN se propone por primera vez en Faster R-CNN [Ren et al., 2017]. Antes del método RPN, los métodos tradicionales de extracción de propuestas requerían mucho tiempo. Por ejemplo, la búsqueda selectiva necesita 2 segundos para procesar una imagen. Además, estas propuestas no son lo suficientemente buenas para la detección. La enumeración de múltiples anclajes y las características de convolución compartida hacen que el método de extracción de la propuesta sea eficiente en el tiempo, a la vez que se logra una alta calidad. RPN es capaz de extraer propuestas más precisas gracias a la supervisión tanto de la clasificación de los antecedentes como de la regresión de los cuadros delimitadores. Existen varias variantes de Faster R-CNN que utilizan RPN. R-FCN [Dai et al., 2016] tiene en cuenta la información de posición del componente y FPN [Kirillov et al., 2019] emplea una red de pirámide de características para mejorar el rendimiento de la detección de objetos pequeños. A diferencia de los detectores de dos etapas, las versiones mejoradas de RPN, como SSD [Liu et al., 2015] y YOLO9000 [Redmon and Farhadi, 2016] son detectores eficientes. RPN tiene muchas aplicaciones exitosas en la detección debido a su velocidad y gran rendimiento, sin embargo, no ha sido plenamente explotado en el seguimiento.

2.1.3. *Faster R-CNN*

Los algoritmos *R-CNN* [Girshick et al., 2014] y *Fast R-CNN* [Girshick, 2015] utilizan la búsqueda selectiva para encontrar propuestas de objetos en una región. Este proceso es algo lento que afecta al rendimiento de la red. Por lo tanto, Shaoqing Ren crea un algoritmo de detección de objetos que no usa el algoritmo de búsqueda selectiva que hace que la red aprenda las propuestas de la región.

Similar a *Fast R-CNN*, *Faster R-CNN* hace que la imagen se proporciona como entrada a una red convolucional, que proporciona un mapa de características convolucional.

En lugar de utilizar un algoritmo de búsqueda selectiva en el mapa de características para identificar las propuestas de la región, se utiliza una red separada para predecir las propuestas de la región. Las propuestas de regiones pronosticadas se remodelan utilizando una capa de agrupación de RoI que se utiliza para clasificar la imagen dentro de la región propuesta y predecir los valores de *offset* para los recuadros delimitadores.

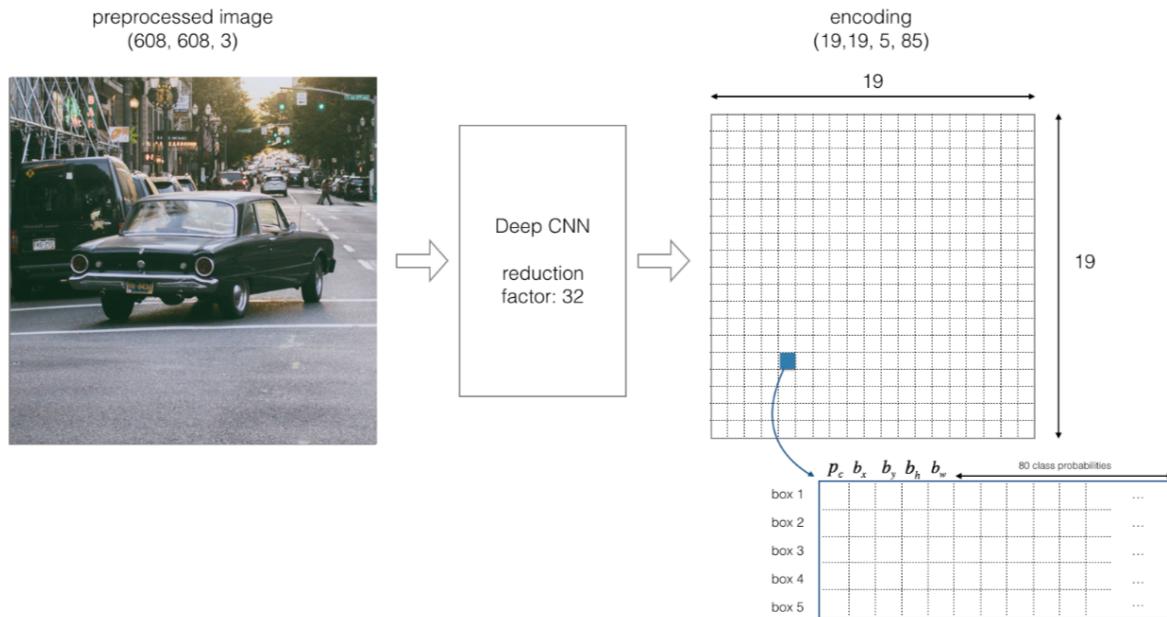


Figura 2.2: Detección con YOLO: Se divide la imagen en celdas, típicamente su cuadrícula de 19×19 . Cada celda será responsable de predecir 5 casillas delimitadoras (en caso de que haya más de un objeto en esta celda). Esto nos dá 1805 bounding boxes con las variables: p_c para el objeto predicho , b_x y b_y para el centro de los bounding boxes, b_h para la altura, b_w para el ancho, para una imagen. Fuente: [Redmon et al., 2016]

2.1.4. YOLO

You Only Look Once (YOLO), es una red convolucional utilizada para realizar la detección de objetos. La tarea de esta detección de objetos es determinar la ubicación de objetos en una imagen, el proceso YOLO se puede apreciar en la Fig. 2.2. Los algoritmos de detección previos reutilizan los clasificadores o localizadores para realizar la detección. Aplican el modelo a una imagen en múltiples ubicaciones y escalas. Las zonas de alta puntuación de la imagen se consideran detecciones. YOLO utiliza un enfoque totalmente diferente. Aplica una sola red neural a la imagen completa. Esta red divide la imagen en regiones y predice cuadros delimitadores y probabilidades para cada región. Estas cajas delimitadoras son ponderadas por las probabilidades pronosticadas. Este tipo de modelo tiene ventajas sobre otros modelos en el proceso de clasificación. El proceso de YOLO se encarga de examinar la imagen completa al hacer pruebas, donde sus predicciones se basan en el contexto global de la imagen. Con YOLO también se puede hacer predicciones con tan solo una evaluación de red, por lo que otros modelos de detección no lo hacen, por ejemplo R-CNN requieren miles de pruebas para una sola imagen. Dado esa característica

de YOLO hace que la detección sea muy rápida, 1000 veces más que Fast R-CNN y 100 veces más que FASTER R-CNN [Hsu et al., 2018]

2.1.5. Asociación de datos

Para el seguimiento de objetos con una probabilidad de detección inferior a la unidad en presencia de falsos positivos (FP), es crucial la asociación de datos (decidir cuál de las múltiples mediciones recibidas se usará para actualizar cada seguimiento). La mayoría de los algoritmos que toman una decisión difícil sobre el origen de la medición real comienzan a fallar a medida que aumenta la tasa de FP o con objetivos de maniobra observables. En lugar de usar solo una medición entre las recibidas y descartar las otras. Se han desarrollado varios algoritmos para resolver este problema [bar shalom and Blair, 2000], [Daum, 1996]. Dos soluciones simples son el filtro de vecino más fuerte (*SNF*) y el filtro de vecino más cercano (*NNF*). En el *SNF*, la señal con la intensidad más alta entre las mediciones validadas se utiliza para la actualización de la pista y las demás se descartan. En el *NNF*, se usa la medida más cercana a la medida predicha. Si bien estas técnicas simples funcionan razonablemente bien con objetivos benignos en escenarios dispersos, comienzan a fallar a medida que aumenta la tasa de FP.

Un enfoque alternativo es usar todas las mediciones validadas con diferentes ponderaciones (probabilidades), conocida como *probabilistic data association (PDA)*. Como tal, los métodos de asociación de datos tienen un fuerte fundamento matemático y son herramientas generales valiosas para los investigadores de la visión por computador [Wang et al., 2018] y estas investigaciones se han extendido en el manejo de múltiples objetos.

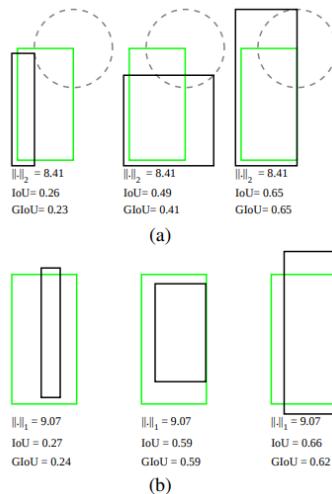


Figura 2.3: Dos conjuntos de ejemplos (a) y (b) con los recuadros delimitadores representados por (a) dos esquinas (x_1, y_1, x_2, y_2) y (b) centro y tamaño (xc, yc, w, h). Para los tres casos en cada conjunto (a) distancia normal L2, $\| \cdot \|_2$, y (b) distancia normal L1, $\| \cdot \|_1$, entre la representación de dos rectángulos son exactamente el mismo valor, pero sus valores *IoU* y *GIoU* son muy diferentes. Fuente: [Bochinski et al., 2017]

2.1.6. *Intersect over Union (IoU)*

IoU, también conocido como índice Jaccard, es la métrica más comúnmente utilizada para comparar la similitud entre dos formas arbitrarias. *IoU* codifica las propiedades de forma de los objetos en comparación, por ejemplo, los anchos, alturas y ubicaciones de dos cuadros delimitadores, en la propiedad *región* y luego calcula una medida normalizada que se centra en sus áreas (o volúmenes). Esta propiedad hace que el *IoU* sea invariable a la escala del problema en cuestión. Debido a esta atractiva propiedad, todas las medidas de rendimiento utilizadas para evaluar la segmentación, la detección de objetos y el seguimiento [Ning et al., 2017] se basan en esta métrica. Sin embargo, puede demostrarse que no existe una fuerte correlación entre la minimización de las pérdidas comúnmente utilizadas, por ejemplo, ln_{norms} , definidas en la representación paramétrica de dos cajas delimitadoras en 2D/3D y la mejora de sus valores de *IoU*. Por ejemplo, considere el escenario 2D simple de la Fig. 2.3 (a), donde la caja delimitadora prevista (rectángulo negro) y la caja de verdad de tierra (rectángulo verde) están representadas por sus esquinas superior izquierda e inferior derecha, es decir, (x_1, y_1, x_2, y_2) .

2.1.7. Filtro de Kalman

El filtro de Kalman se sitúa dentro de los problemas de filtrado, que quedan englobados en la teoría de procesos estocásticos. A que nos referimos cuando decimos que nos adentramos en los modelos estocásticos? La necesidad de ir más allá de los sistemas deterministas -aquellos en los que el azar no interviene- se hace patente ante sus deficiencias, las cuales vamos a intentar evidenciar brevemente. Dado un fenómeno físico, lo intentamos modelizar a partir de sus comportamientos, y lo haremos utilizando leyes físicas conocidas u observaciones recogidas, las cuales relacionaremos entre sí para obtener una salida del sistema. Sin embargo, un sistema determinista no es suficiente para llevar a cabo este análisis. Las razones son diversas. Por un lado, un modelo matemático acaba por recoger solamente aquellas características dominantes, por lo que muchos efectos quedan fuera del modelo. Por otro lado, los sistemas dinámicos no quedan definidos simplemente por los efectos que recogemos, sino que también existen ruidos que no podemos modelar de forma determinista [Basar, 2001].

2.1.8. Método Húngaro

El método húngaro es un algoritmo que se utiliza en problemas de asignación cuando se quiere minimizar el costo. Es decir, se usa para encontrar el costo mínimo al asignar varias personas a diversas actividades basadas en el menor costo. Se debe asignar cada actividad a una persona diferente.

Un problema de asignación es un tipo especial de problema de programación lineal, donde el objetivo es minimizar el costo o el tiempo de completar una cantidad de trabajos por parte de varias personas. Una de las características importantes del problema de asignación es que solo se asigna un trabajo (o trabajador) a una máquina (o proyecto) [Kuhn, 1955].

Capítulo 3

Estado del Arte

En el siguiente capítulo se desarrolla los métodos para el proceso de seguimiento y detección de objetos explorando colecciones de publicaciones científicas.

Para ello agrupamos los métodos existentes en dos categorías según el contenido utilizado: detección y seguimiento de objetos.

3.1. Detección de objetos

La detección de objetos tiene como objetivo localizar y clasificar objetos existentes en cualquier imagen, y etiquetarlos con cuadros rectangulares para mostrar las confidencias de la existencia. Aunque los CNN han obtenido un excelente rendimiento en la detección de objetos genéricos [Liu et al., 2018], ninguno de estos enfoques ha logrado mejores resultados que el mejor método artesanal basado en características [BuSS et al., 2018] durante mucho tiempo, incluso cuando se incorporan la información basada en la pieza y el manejo de la oclusión. Por lo tanto, se han llevado a cabo algunas investigaciones para analizar las razones. [Zhang et al., 2014] intentaron adaptar el genérico Faster R-CNN a la detección de peatones [Hailong Li et al., 2016]. Modificaron el clasificador aguas abajo añadiendo bosques potenciados a los mapas de características de convección compartidos de alta resolución y tomando un RPN para manejar instancias pequeñas y ejemplos negativos duros. Para tratar las oclusiones complejas que existen en las imágenes de peatones, inspiradas en *Deformable Part-based Model* (DPM) [Mordan et al., 2017]. [Ali et al., 2016] propusieron un marco de aprendizaje profundo llamado DeepParts, que toma decisiones basado en un conjunto de detectores de partes extensas. DeepParts tiene ventajas en el manejo de datos mal etiquetados, propuestas positivas de IoU bajas y oclusión parcial. Otros investigadores también intentaron combinar información complementaria de múltiples fuentes de datos. CompACT-Deep adopta una cascada consciente de la complejidad para combinar características artesanales y CNNs afinados [He et al., 2014]. Basándose en R-CNN más rápido, [Liu et al., 2015] propusieron redes neuronales profundas multiespectrales para la detección de peatones con el fin de combinar información complementaria de color e imágenes térmicas [Jung and Lyou, 2015]. A pesar de ser buenos algoritmos, no son muy usados en el ámbito de seguimiento de objetos en tiempo real.

[Tian et al., 2017] propusieron un asistente de tareas de CNN (TA-CNN) para aprender conjuntamente múltiples tareas con múltiples fuentes de datos y para combinar los atributos de peatones con los atributos semánticos de la escena. Du et al. propusieron una arquitectura de fusión de red neural profunda para una detección rápida y robusta de peatones. Basándose en las cajas delimitadoras candidatas generadas con detectores de *Single Shot MultiBox Detector* (SSD) [Liu et al., 2015], se procesan paralelamente múltiples clasificadores binarios para llevar a cabo la fusión de red basada en el rechazo blando *Soft-rejection based Network Fusion* (SNF) consultando su grado de confianza agregado. Sin embargo, la mayoría de estos enfoques son mucho más sofisticados que el marco estándar R-CNN. CompACT-Deep consta de una variedad de características artesanales, un modelo pequeño de CNN y un modelo grande de VGG16 [Selimovi et al., 2018]. DeepParts contiene 45 modelos CNN afinados, y se requiere un conjunto de estrategias, incluyendo el manejo de cambio de caja delimitadora y la selección de partes, para llegar a los resultados reportados. Por lo tanto, la modificación y simplificación es importante para reducir la carga tanto del software como del hardware para satisfacer la demanda de detección en tiempo real. [Tome et al., 2016] propusieron una solución novedosa para adaptar la tubería genérica de detección de objetos a la detección de peatones mediante la optimización de la mayoría de sus etapas. [Hu et al., 2018] formaron un conjunto de modelos de decisión mejorados reutilizando los mapas de características convincentes, y se obtuvo una mejora adicional con el simple etiquetado de píxeles y otras características complementarias hechas a mano. [Tome et al., 2017] propusieron una arquitectura de CNN profunda basada en una región de memoria reducida, que fusiona las respuestas regionales tanto de los detectores *Aggregate Channel Features* (ACF) como de los clasificadores *Support Vector Machine* (SVM) en R-CNNN. [Ribeiro et al., 2018] abordaron el problema de la Navegación Humana y propusieron un sistema de seguimiento de personas basado en la visión y guiado por múltiples sensores de cámara.

3.2. Seguimiento de Objetos

El vasto cuerpo de investigación sobre el seguimiento de objetos contiene variaciones en técnicas hechas por las figuras notables del seguimiento multiobjeto como: BarShalom, Reid y Blackman, sus técnicas clásicas como *Probabilistic Data Association* (PDA) y *Multiple Hypothesis Tracking* (MHT) [Musicki et al., 1994] han influido en generaciones de investigadores de SSA en la creación de técnicas como *Joint Probabilistic Data Association* (JPDA), *TrackOriented MHT* (TOMHT), y *Hypothesis-Oriented MHT* (HOMHT) [Streit, 2016]. [Milan et al., 2018] propusieron un framework basado en campos aleatorios condicionales, en el que modelaron exclusiones a nivel de detección y a nivel de pista para mejorar el rendimiento del rastreo en situaciones de aglomeraciones. A este método se le hizo una mejora obteniendo un conjunto de pistas y luego se diseñó una energía discreta-continua para reconstruir las trayectorias finales. Sin embargo, el rendimiento de estos dos enfoques se basó en la calidad de las propuestas de las pistas y en el seguimiento de los errores en las pistas iniciales propagados a los resultados finales. [Enach et al., 2016] propusieron que debido a las detecciones dentro de este *framework* son consideradas como puntos en el proceso de seguimiento, trae consigo ambigüedades en la asociación de datos, especialmente en situaciones de hacinamiento. Para hacer frente a este problema, ampliaron el enfoque de MHT mediante la incorporación de un novedoso modelo de

mejora de la detección que incluía el análisis de la escena de detección y el análisis de detección-detección; el primero modela la escena mediante el uso de densas detecciones confidenciales y maneja las trayectorias falsas, mientras que el segundo estima las correlaciones entre las detecciones individuales y mejora la capacidad de hacer frente a las hipótesis de objetos cercanos en situaciones de hacinamiento. [Toin at al., 2018] evaluaron un *Joint Probabilistic Data Association Filter* (JPDAF) original de Monte Carlo para el seguimiento de objetivos autónomos interactivos en un entorno desordenado. La originalidad del algoritmo propuesto consiste en reducir la complejidad del paso de predicción seleccionando y actualizando por separado los grupos de objetivos en interacción. La complejidad del paso de corrección es tratada por DA y un procedimiento de puerta como se encuentra en la literatura. Los principales supuestos que hicieron en este trabajo son (i) que la evolución del estado de cada objetivo sólo depende de los estados de todos los objetivos en el paso anterior y (ii) que un simulador genérico o una función que modele los comportamientos de los objetivos y sus interacciones mutuas está disponible. También construyeron un gráfico de interacción aproximada entre objetivos sobre la marcha sobre la base de información simple como su ubicación, como se ha hecho en trabajos anteriores. MHT y JPDAF, ambos métodos han sido revisados recientemente en un escenario de seguimiento por detección y han mostrado resultados prometedores [Funk et al., 2017]. Sin embargo, el rendimiento de estos métodos aumenta la complejidad computacional y de implementación.

Capítulo 4

Propuesta

En el Capítulo 3 se repasa los métodos que son utilizados para la detección y el seguimiento de objetos. La propuesta combina algunas de estos métodos para el seguimiento multi-objeto. Un reto importante en el seguimiento por detección es cómo asociar los resultados de detección poco fiables con las pistas existentes, y todo este proceso ejecutarlo en tiempo real. En esta tesis, proponemos manejar la detección poco fiable mediante la recopilación de candidatos de los resultados de la detección y el seguimiento, reemplazando el modelo de detección base *Faster-RCNN* por Yolov2. Y una mejora en la métrica de asociación de datos dando una mejor asignación de posibles pistas en el proceso de seguimiento, este método hace referencia a la investigación [Wojke et al., 2017], con el uso de técnicas que aplican modelos de detección RPN para la re-identificación de objetos en cada *frame*.

La propuesta se compone de dos pasos principales: (i) detectar la ubicación de los objetos de forma independiente en cada *frame*, (ii) seguimiento de objetos formando pistas, vinculando las detecciones correspondientes a lo largo del tiempo. (Ver Figura 4.2).

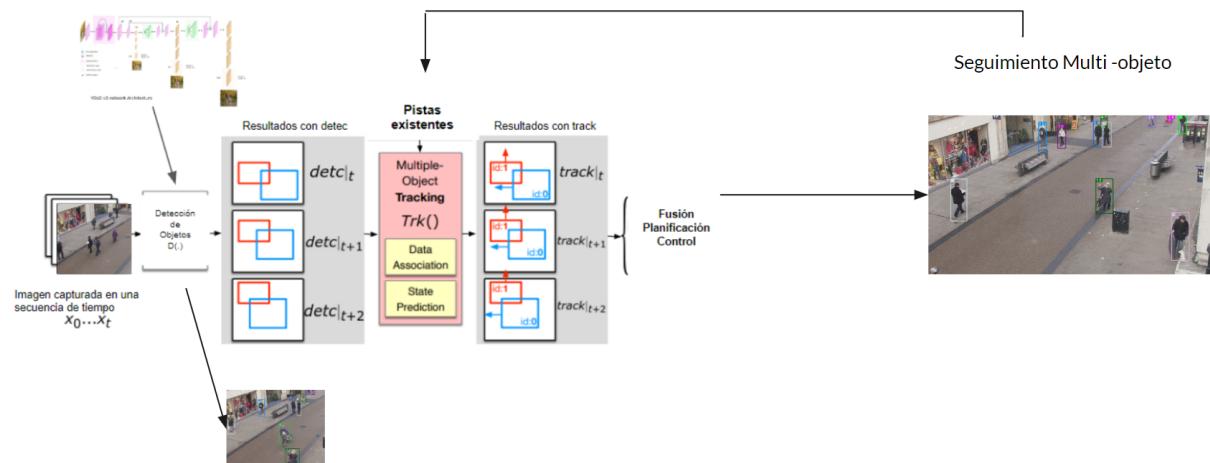


Figura 4.1: Pipeline. El canal completo de percepción visual en la conducción autónoma, es decir, la detección de objetos y Seguimiento de objetos múltiples (MOT).

4.1. Detección de Objetos

El elemento central del nuevo método de seguimiento es un detector basado en regresión. Es decir, se entrena una YOLO en el conjunto de datos de detección de peatones MOT17Det [Bergmann et al., 2019]. Para realizar la detección de objetos. YOLO aplica una RPN para generar una multitud de propuestas de cajas delimitadoras para cada objeto potencial. Los mapas de características de cada propuesta se extraen a través de la agrupación de regiones de interés y se transmiten a los responsables de la clasificación y la regresión. El cabezal de clasificación asigna una puntuación de objeto a la propuesta, en este caso, se evalúa la probabilidad de que la propuesta muestre un peatón. La cabeza de regresión refina la ubicación de la caja delimitadora alrededor de un objeto. El detector produce el conjunto final de detecciones de objetos mediante la aplicación de la no supresión del máximo (NMS) a las propuestas de recuadros delimitadores refinados. El método que presento aprovecha la capacidad antes mencionada de retroceder y clasificar las cajas delimitadoras para realizar un seguimiento multiobjeto. (Ver Figura 4.2).

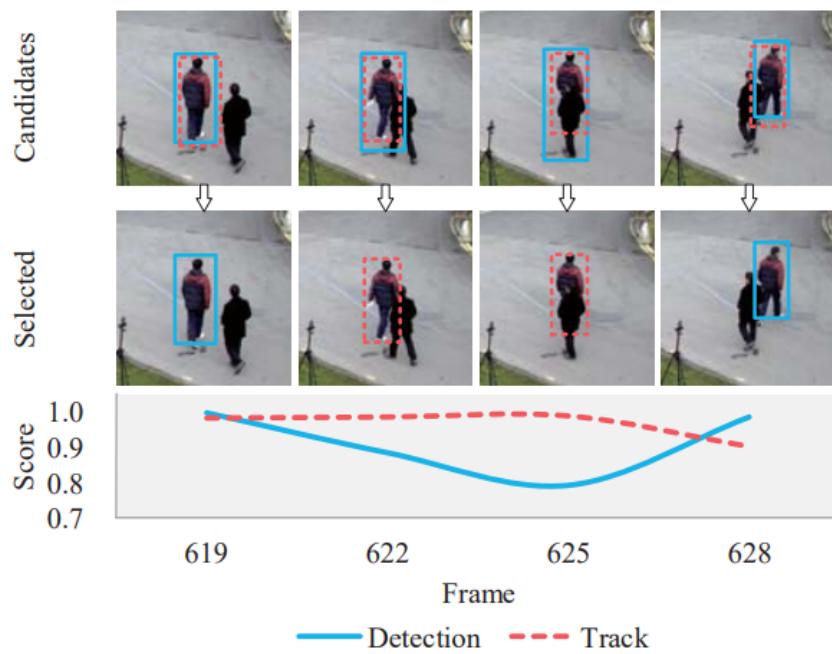


Figura 4.2: Selección de candidatos en base a puntuaciones unificadas. Los candidatos de la detección y las pistas se visualizan como rectángulos sólidos azules y rectángulos punteados rojos, respectivamente. La detección y las pistas pueden complementarse entre sí para la asociación de datos. Fuente: [Xu et al., 2019].

La CNN que YOLO utiliza se inspira en el modelo GoogLeNet [Aswathy et al., 2018] que introduce los módulos de inicio. La red tiene 24 capas convolucionales seguidas de 2 capas completamente conectadas. Las capas de reducción con filtros de 1x1, seguidas de capas convolucionales de 3x3 sustituyen a los módulos iniciales. El modelo *YOLOv3* en la Figura 4.3 está enfocado en mejorar la precisión sin dejar de ser un detector rápido. La normalización de lotes se añade para evitar el sobre equipamiento sin necesidad de utilizar la función *dropout*. Se aceptan imágenes de mayor resolución: el modelo YOLO utiliza imágenes de 448x448, mientras que el YOLOv2 utiliza imágenes de 608x608, lo que

permite la detección de objetos potencialmente más pequeños.

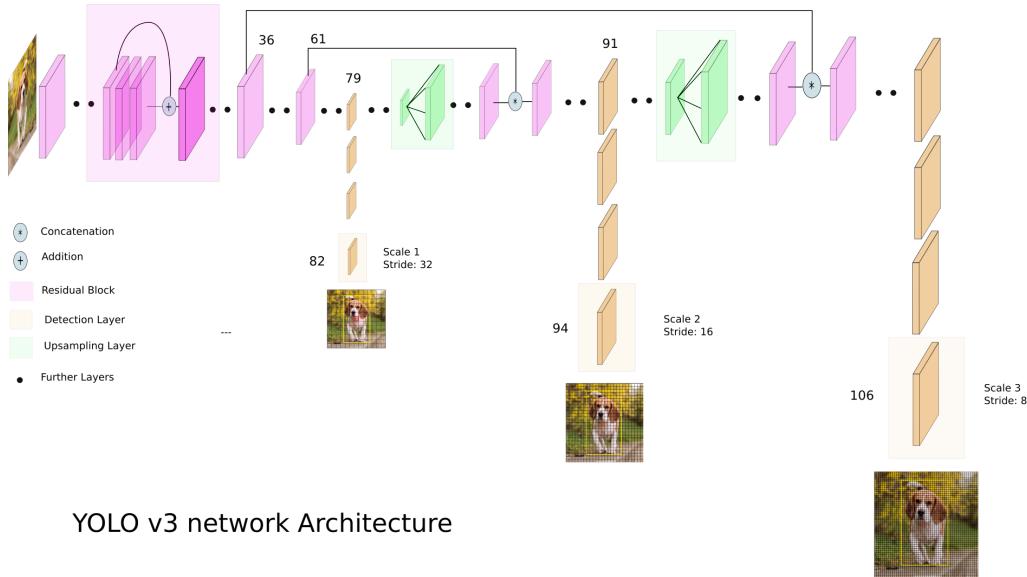


Figura 4.3: Arquitectura YOLOv3: se compone de 24 capas convolucionales y 2 capas totalmente conectadas. Fuente: [J. Redmon et al., 2016].

Con los modelos Faster-RCNN y SSD, las cajas delimitadoras previstas a menudo contenían un objeto. Sin embargo, el modelo YOLO predice un gran número de cajas delimitadoras. Por lo tanto, hay muchas cajas delimitadoras sin ningún objeto. El método de supresión no máxima (NMS) se aplica al final de la red. Consiste en fusionar cajas delimitadoras altamente superpuestas de un mismo objeto en una sola. Los autores observaron que todavía hay pocos falsos positivos detectados.

Es necesario recalcar que las SSD se basa en la exactitud lo que es una buena razón para usarlo pero, YOLO es la mejor manera de seguir adelante. En primer lugar, una reflexión visual de la rapidez frente a un compromiso de precisión los diferenciaría bien, esto se puede ver en la Tabla 4.1. YOLO tiene limitación en cuanto al detectar objetos pequeño y visto la comparación con los demás detectores, YOLO es el menos preciso por ello se creó el modelo YOLOv2 [Redmon and Farhadi, 2017] que está enfocado en mejorar la precisión sin dejar de ser un detector rápido. La normalización de lotes se añade para evitar el sobre equipamiento sin necesidad de utilizar la función "dropout". Se aceptan imágenes de mayor resolución: el modelo YOLO utiliza imágenes de 448x448, mientras que el YOLOv2 utiliza imágenes de 608x608, lo que permite la detección de objetos potencialmente más pequeños.

Entonces YOLO es más rápido en comparación con Faster-RCNN y SSD. Sus precisiones son comparativamente similares. YOLO no funciona muy bien para objetos pequeños, pero se puede mejorar su rendimiento con la versión 2 para objetos más pequeños aumentando el número de cajas de anclaje y disminuyendo el umbral de IoU.

Model	# mAP	# FPS	# Real Time speed
Fast YOLO	52.7 %	155	Yes
YOLO	63.4 %	45	Yes
YOLO VGG-16	66.4 %	21	No
Faster R-CNN	70.0 %	0.5	No
Faster R-CNN VGG-16	73.2 %	7	No
Faster R-CNN ZF	62.1 %	18	No

Tabla 4.1: Sistemas en tiempo real sobre PASCAL VOC 2007. Comparación de velocidades y rendimientos para los modelos entrenados con los conjuntos de datos PASCAL VOC 2007 y 2012. Los resultados publicados corresponden a las implementaciones de [J. Redmon et al. 2016].

4.2. Seguimiento de Objetos

La asociación de datos es esencial no sólo en el momento de la inferencia, donde las detecciones deben estar asociadas a diferentes pistas, sino también en el momento de la evaluación (evaluación del entrenamiento y del rendimiento), donde las pistas inferidas deben estar asociadas a la realidad del terreno.

La precisión y exactitud del seguimiento de objetos múltiples (*Multiple Object Tracking Accuracy* (MOTA) y *Multiple Object Tracking Precision* (MOTP)) son dos métricas estándar y ampliamente utilizadas para evaluar la calidad de los rastreadores de objetos múltiples. Están diseñados específicamente para codificar los desafíos y dificultades de rastrear múltiples objetos.

Se propone un proxy diferenciable para el MOTA y la MOTP, lo que permite entrenar un rastreador profundo de objetos múltiples mediante la optimización directa de las métricas estándar de la MOT. La aproximación propuesta se basa en una red bidireccional recurrente que introduce la matriz de distancia objeto-hipótesis y produce la asociación óptima de hipótesis-objeto, emulando así el algoritmo húngaro. Seguido de un módulo diferenciable, la asociación estimada se utiliza para calcular la MOTA y la MOTP. La red recurrente bidireccional es llamada *Deep Hungarian Network* (DHN). Una vez entrenado, el DHN puede utilizarse para proporcionar una aproximación de la asignación óptima de pista a el objetivo..

4.2.1. Problema de asignación

Una forma convencional de resolver la asociación entre los estados de Kalman predichos y las mediciones recién llegadas es crear un problema de asignación que se pueda resolver utilizando el algoritmo húngaro. En esta formulación de problemas, integramos la información de movimiento y apariencia mediante la combinación de dos métricas apropiadas. Para incorporar información de movimiento, usamos la distancia (cuadrada) de Mahalanobis [Cen et al., 2012] entre los estados de Kalman predichos y las mediciones recién llegadas.

En combinación con los datos que muestra la detección con YOLO, la métrica Mahalanobis se complementa para servir diferentes aspectos del problema de asignación. Por un lado, la distancia de Mahalanobis proporciona información sobre posibles ubicaciones de objetos basadas en el movimiento que son particularmente útiles para las predicciones a corto plazo junto la distancia del coseno considera información de apariencia que es particularmente útil para recuperar identidades después de occlusiones a largo plazo, cuando el movimiento es menos discriminatorio. Por otro lado se construye el problema de asociación, combinamos ambos modelos. Primero denotar Mahalanobis 4.1

$$d^{(1)}(i, j) = (d_j - y_i)^T S_i^{-1} (d_j - y_i) \quad (4.1)$$

donde denotaremos la proyección de la distribución de la pista i -ésima en el espacio de medición por (y_i, S_i) y la detección de la caja delimitadora j -ésima por d_j . La distancia de Mahalanobis tiene en cuenta la incertidumbre de la estimación del estado midiendo cuántas desviaciones estándar se aleja la detección de la ubicación media de la pista.

Luego denotamos la distancia del coseno 4.2:

$$d^{(2)}(i, j) = \min\{1 - r_j^T r_k^{(i)} | r_k^{(i)} \in R_i\} \quad (4.2)$$

donde para cada d_j de detección de cajas delimitadoras se calcula un descriptor de apariencia r_j con $\|r_j\| = 1$. Además, mantenemos una galería $R_k = \{r_{(i)}^k\}_{k=1}^{L_k}$ de la última $L_k = 100$ descriptores de apariencia asociados para cada pista k . Entonces, nuestra segunda métrica mide la menor distancia coseno entre la i -ésima pista y la j -ésima detección en el espacio de apariencia.

Por último la combinación de ambas métricas se complementan entre sí sirviendo a diferentes aspectos del problema de la asignación. Por un lado, la distancia de Mahalanobis proporciona información sobre posibles ubicaciones de objetos basados en el movimiento que son particularmente útiles para las predicciones a corto plazo. Por otro lado, la distancia del coseno considera información de apariencia que es particularmente útil para recuperar identidades después de occlusiones a largo plazo, cuando el movimiento es menos discriminatorio. Para construir el problema de la asociación se combina ambas métricas utilizando una suma ponderada:

$$c_{(i,j)} = \lambda d^{(1)}(i, j) + (1 - \lambda)d^{(2)}(i, j) \quad (4.3)$$

Donde la influencia de cada métrica en el costo combinado de la asociación puede ser controlada a través del hiperparámetro λ . Durante los experimentos se aconseja usar el ajuste $\lambda = 0$, esta es una elección razonable cuando hay un movimiento sustancial de la cámara. En este ajuste, sólo se utiliza la información de apariencia en el término de costo de asociación.

Capítulo 5

Pruebas y resultados

En el capítulo anterior se realizó una descripción de la propuesta y el detalle de sus componentes. En este capítulo se describen las pruebas y resultados obtenidos a partir de la implementación de la propuesta y los estudios realizados. Primero se especifica los conjuntos de datos y posteriormente los resultados.

5.1. Conjuntos de datos

Los conjuntos de datos son extraídos de la página de *MOTBenchmark* [Luo et al., 2017]. Éste nos ofrece un conjunto de videos y una herramienta *DevKit* para el cálculo de las métricas 5.1. Adicionalmente se obtuvo una base de datos brindado por la Universidad Católica San Pablo para realizar pruebas y obtener resultados cualitativos. Las características de los videos obtenidos para las pruebas se reflejan en la tabla 5.2.

5.2. Resultados

Para un correcto análisis del uso de YOLO como detector de la propuesta, se tuvo que comparar el rendimiento del *Tracking* mediante el intercambio de la propuesta y el

Métrica	Mejor	Descripción
MOTA	alto	<i>Accuracy.</i> Esta medida combina tres fuentes de error: falsos positivos, objetivos perdidos y cambios de identidad.
MOTP	alto	<i>Precision.</i> La desalineación entre los recuadros delimitadores anotados y los previstos.
MT	alto	La mayoría de los objetivos rastreados. La proporción de trayectorias de tierra-verdaderas que están cubiertas por una hipótesis de vía durante al menos el 80 por ciento de su vida útil respectiva.
Hz	alto	Velocidad de procesamiento (en cuadros por segundo, excluyendo el detector).

Tabla 5.1: Métricas del *MOT16* y *MOT17* [Luo et al., 2017].

Data	FPS	Resolution	Tiempo
MOT16/MOT17/MOT20	14-30	1920x1080/640X480	(00:15)/(01:25) minutos
UCSP	30	1920x1080	(01:25) minutos

Tabla 5.2: Características de los 2 conjuntos de datos utilizados en las pruebas.

modelo base. Estos resultados son evaluados con las secuencias de validación del MOT20 Tabla 5.3 .

Se evaluó el rendimiento de la técnica base DeepSORT, junto a la propuesta *Deep-SORTYOLO*. Dado el conjunto diverso de videos de pruebas: tabla 5.4, tabla 5.5, tabla 5.4, tal y como se establece en la base de datos MOT, y las secuencias de video que la UCSP ofrece. La arquitectura de detección utilizada para la técnica base es la Faster-RCNN y para la técnica propuesta se usa YOLO.

Dado que es difícil utilizar una sola puntuación para evaluar el rendimiento del seguimiento de objetivos múltiples, utilizamos las métricas de evaluación definidas en la tabla 5.1 dadas por el *benchmark MOT*. Para la evaluación con los datos de la cámara del pasaje de la UCSP sólo se pudo sacar resultados cualitativos.

Rastreador	Detector	Detección/Precisión	Tracking/MOTA
SORT	FrRCNN	64.4	33.5
SORT	YOLOv2	64.1	24.8
SORTYO	FrRCNN	65.12	32.9
SORTYO	YOLOv2	65.2	24.6

Tabla 5.3: Comparación del rendimiento del *Tracking* mediante el intercambio de las propuestas y el modelo base del. Estos resultados son evaluados con las secuencias de validación del *MOT20*

Benchmark	Method	MT	MOTA	MOTP	Hz
MOT16	DeepSORT	41	21.9	77.9	3.0
MOT16	DeepSORTYO	39	25.2	75.6	4.1

Tabla 5.4: Resultados cuantitativos de las líneas de base en *MOT16* con datos de prueba [Luo et al., 2017].

5.2.1. Evaluación del Desempeño

Con el rendimiento de YOLO en la comparativa 5.3 se puede rescatar que YOLO en precisión tiene mayor resultado a las comparativas. Pero un *accuracy* menor al modelo base. Donde en la investigación de la tabla 4.1 se puede inferir que deberá tener una mejor velocidad.

Con el rendimiento del seguimiento con la base de datos *benchmark MOT*, la Tabla 5.4 nos muestra que *DeepSORTYO* da un mejor resultado en la métrica *MOTA* que

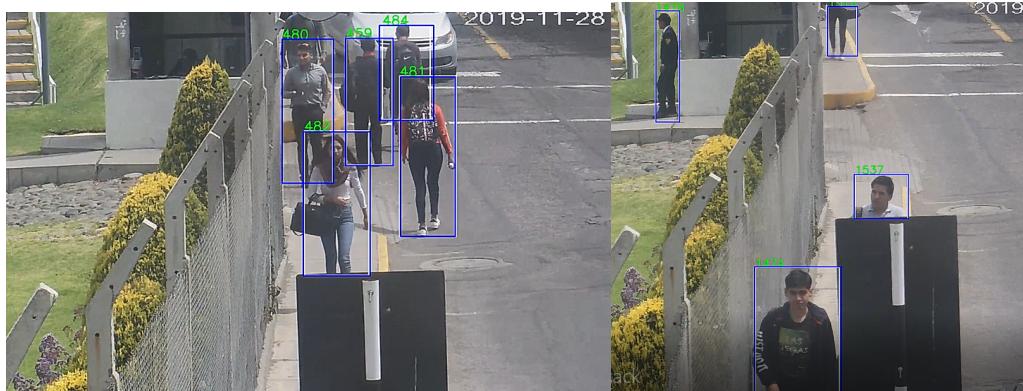


Figura 5.1: Buenos resultados cualitativos



Figura 5.2: Mal resultado cualitativo

Figura 5.3: Secuencia del *tracking* multi-objeto, usando la base de datos UCSP

DeepSORT con las secuencias MOT16, pero en la evaluación de la métrica *MOTP* el modelo propuesto no llega a superar al modelo base. Los *Hz* si dan un resultado favorable, debido al nuevo detector propuesto. El concepto de cada métrica se puede apreciar en la Tabla 5.1. Luego se evaluó los mismos métodos pero con una base de datos MOT17 y MOT20, estos resultados se reflejan en la Tabla 5.5, y Tabla 5.6 donde el método *DeepSORTYO* logra un mejor *Hz* en ambas base de datos, pero solo mejora en *MT* en el MOT20. Las métricas MOTA Y MOTP aun mantiene un resultado debajo al modelo base. La comparativa de los métodos en video se en la Figura 5.6.

Y con respecto al seguimiento con la base de datos UCSP, estos se reflejan en la figura 5.3. El resultado muestra un buen rendimiento cuando los objetos están a una

Benchmark	Method	MT	MOTA	MOTP	Hz
MOT17	DeepSORT	38	29.4	77.9	2.8
MOT17	DeepSORTYO	34	27.2	75.6	4.0

Tabla 5.5: Resultados cuantitativos de las líneas de base en *MOT17* con datos de prueba [Luo et al., 2017].



Figura 5.4: Técnica DeepSORT



Figura 5.5: Técnica DeepSORTYO

Figura 5.6: Secuencia del *tracking* multi-objeto, usando la base de datos *MOT17-04*

Benchmark	Method	MT	MOTA	MOTP	Hz
MOT20	DeepSORT	36	26.2	77.9	2.9
MOT20	DeepSORTYO	39	22	69.6	3.5

Tabla 5.6: Resultados cuantitativos de las líneas de base en *MOT20* con datos de prueba [Luo et al., 2017].

distancia no muy corta como se ve en la figura 5.1, y cuando los objetos tienden a reducir su tamaño debido a occlusiones como se ve en la figura 5.2.

5.2.2. Tiempo de Ejecución

La mayoría de las soluciones *MOT* tienen como objetivo impulsar el rendimiento hacia una mayor precisión, a menudo, a costa del rendimiento en tiempo de ejecución. El rendimiento en tiempo real es esencial, donde algunos métodos que logran la mejor precisión también tienden a ser los más lentos. *DeepSORTYOLO* combina las dos propiedades deseables, velocidad y precisión.

Capítulo 6

Conclusiones y Trabajos Futuros

En el anterior capítulo se demostró los resultados y un análisis de los diferentes experimentos. En este capítulo se exponen las conclusiones, limitaciones o problemas encontrados, recomendaciones y trabajos futuros.

Se propone un modelo que extiende el paradigma actual para el seguimiento de objetos. Los experimentos y análisis del modelo base con la propuesta, prueban una pequeña mejora en cuanto a la métrica de velocidad *Hz*. Con la inclusión del componente de detección YOLO, la efectividad en la obtención de resultados se incrementó, pues permitió obtener desde un inicio una detección más precisa para el seguimiento.

En cuanto a los aspectos relacionados al *accuracy*, se argumenta que para este tipo de base de datos usados es necesario un entrenamiento más exhaustivo al momento de la implementación, y la modificación de los componentes del modelo de re-identificación.

6.1. Problemas encontrados

Modificar un re-detector en el seguimiento implicó que muchas veces el *accuracy* disminuyera y debido a que el modelo propuesto está enfocado para la detección de objetos en video a color con un rango de secuencias de 14 a 30 FPS, la detección de los objetos con la base de datos de la UCSP, que son a una escalar mayor a 30 PFS, está limitada por este tipo conjunto de datos y también al tipo de *hardware* que se utilizó en el modelo base. En otros términos, realizar el seguimiento con la base de datos de la universidad, es posible pero con resultados no esperados en cuanto a la cantidad de FPS al momento de hacer el seguimiento. Esto ocurre porque la herramienta requiere un entrenamiento de más épocas con la base de datos de la UCSP, y así obtener detecciones familiarizadas para que YOLO detecte, y el modelo de seguimiento cumpla con su objetivo.

6.2. Trabajos futuros o recomendaciones

Como consecuencia de la manipulación del método base para conseguir una propuesta, en el nuevo modelo de seguimiento identificamos diferentes aspectos que podrían mejorar el rendimiento. Estos aspectos van desde temas técnicos como el *hardware* usado o hasta la cantidad de *data* para el entrenamiento del modelo.

En relación a la detección propiamente dicha, se recomienda usar más características que nos proporciona YOLO, hacer que los *boxes* detectados con sus respectivas etiquetas aporten más información para el seguimiento.

Finalmente en cuanto al componente seguimiento, el hecho de incluir un re-detector que ofrece mayor características generaría una mayor procesamiento. Lo que conlleva a investigar nuevas propuestas para la re-identificación que ayudará al seguimiento.

Bibliografía

- [Aloysius and Geetha, 2017] Aloysius, N. and Geetha, M. (2017). A review on deep convolutional neural networks. In *2017 International Conference on Communication and Signal Processing (ICCSP)*, pages 0588–0592.
- [Aswathy et al., 2018] Aswathy, P., Siddhartha, and Mishra, D. (2018). Deep googlenet features for visual object tracking. In *2018 IEEE 13th International Conference on Industrial and Information Systems (ICIIS)*, pages 60–66.
- [bar shalom and Blair, 2000] bar shalom, Y. and Blair, W. (2000). Multitarget-multisensor tracking: Applications and advances-volume iii. II.
- [Basar, 2001] Basar, T. (2001). *A New Approach to Linear Filtering and Prediction Problems*, pages 167–179.
- [Bergmann et al., 2019] Bergmann, P., Meinhardt, T., and Leal-Taixé, L. (2019). Tracking without bells and whistles. *CoRR*, abs/1903.05625.
- [Bertinetto et al., 2016] Bertinetto, L., Valmadre, J., Henriques, J. F., Vedaldi, A., and Torr, P. H. (2016). Fully-convolutional siamese networks for object tracking.
- [Bochinski et al., 2017] Bochinski, E., Eiselein, V., and Sikora, T. (2017). High-speed tracking-by-detection without using image information. In *International Workshop on Traffic and Street Surveillance for Safety and Security at IEEE AVSS 2017*, Lecce, Italy.
- [BuSS et al., 2018] BuSS, M., Steiniger, Y., Benen, S., Kraus, D., Kummert, A., and Stiller, D. (2018). Hand-crafted feature based classification against convolutional neural networks for false alarm reduction on active diver detection sonar data.
- [Chau et al., 2013] Chau, D. P., Brémond, F., and Thonnat, M. (2013). Object tracking in videos: Approaches and issues. *CoRR*, abs/1304.5212.
- [Dai et al., 2016] Dai, J., Li, Y., He, K., and Sun, J. (2016). R-FCN: object detection via region-based fully convolutional networks. *CoRR*, abs/1605.06409.
- [Daum, 1996] Daum, F. (1996). Multitarget-multisensor tracking: Principles and techniques [book review]. *IEEE Aerospace and Electronic Systems Magazine*, 11(2):41–.
- [Fan et al., 2019] Fan, J., Ma, C., and Zhong, Y. (2019). A selective overview of deep learning. *ArXiv*, abs/1904.05526.
- [Fiaz et al., 2018] Fiaz, M., Mahmood, A., and Jung, S. K. (2018). Tracking noisy targets: A review of recent object tracking approaches. *CoRR*, abs/1802.03098.

- [Funk et al., 2017] Funk, C., Lee, S., Oswald, M. R., Tsogkas, S., Shen, W., Cohen, A., Dickinson, S., and Liu, Y. (2017). 2017 iccv challenge: Detecting symmetry in the wild. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 1692–1701.
- [Girshick, 2015] Girshick, R. (2015). Fast r-cnn. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448.
- [Girshick et al., 2014] Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587.
- [Granström and Baum, 2016] Granström, K. and Baum, M. (2016). Extended object tracking: Introduction, overview and applications. *CoRR*, abs/1604.00970.
- [Hailong Li et al., 2016] Hailong Li, Zhendong Wu, and Jianwu Zhang (2016). Pedestrian detection based on deep learning model. In *2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 796–800.
- [He et al., 2014] He, K., Zhang, X., Ren, S., and Sun, J. (2014). Spatial pyramid pooling in deep convolutional networks for visual recognition. *CoRR*, abs/1406.4729.
- [He et al., 2017] He, Q., Wu, J., Yu, G., and Zhang, C. (2017). SOT for MOT. *CoRR*, abs/1712.01059.
- [Hsu et al., 2018] Hsu, S., Huang, C., and Chuang, C. (2018). Vehicle detection using simplified fast r-cnn. In *2018 International Workshop on Advanced Image Technology (IWAIT)*, pages 1–3.
- [Jung and Lyou, 2015] Jung, H. and Lyou, J. (2015). Matching of thermal and color images with application to power distribution line fault detection. In *2015 15th International Conference on Control, Automation and Systems (ICCAS)*, pages 1389–1392.
- [Kirillov et al., 2019] Kirillov, A., Girshick, R. B., He, K., and Dollár, P. (2019). Panoptic feature pyramid networks. *CoRR*, abs/1901.02446.
- [Kuhn, 1955] Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(12):83–97.
- [Liu et al., 2018] Liu, L., Ouyang, W., Wang, X., Fieguth, P. W., Chen, J., Liu, X., and Pietikäinen, M. (2018). Deep learning for generic object detection: A survey. *CoRR*, abs/1809.02165.
- [Liu et al., 2015] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S. E., Fu, C., and Berg, A. C. (2015). SSD: single shot multibox detector. *CoRR*, abs/1512.02325.
- [Lu et al., 2019] Lu, Z., Chan, K., Pu, S., and Porta, T. L. (2019). Crowdvision: A computing platform for video crowdprocessing using deep learning. *IEEE Transactions on Mobile Computing*, 18(7):1513–1526.
- [Luo et al., 2017] Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W., Zhao, X., and Kim, T.-K. (2017). Multiple object tracking: A literature review.

- [Milan et al., 2016] Milan, A., Leal-Taixé, L., Reid, I. D., Roth, S., and Schindler, K. (2016). MOT16: A benchmark for multi-object tracking. *CoRR*, abs/1603.00831.
- [Mordan et al., 2017] Mordan, T., Thome, N., Cord, M., and Hénaff, G. (2017). Deformable part-based fully convolutional network for object detection. *CoRR*, abs/1707.06175.
- [Musicki et al., 1994] Musicki, D., Evans, R., and Stankovic, S. (1994). Integrated probabilistic data association. *IEEE Transactions on Automatic Control*, 39(6):1237–1241.
- [Ning et al., 2017] Ning, X., Zhu, W., and Chen, S. (2017). Recognition, object detection and segmentation of white background photos based on deep learning. In *2017 32nd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pages 182–187.
- [Reddy et al., 2015] Reddy, K. R., Priya, K. H., and Neelima, N. (2015). Object detection and tracking – a survey. In *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, pages 418–421.
- [Redmon et al., 2016] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788.
- [Redmon and Farhadi, 2016] Redmon, J. and Farhadi, A. (2016). YOLO9000: better, faster, stronger. *CoRR*, abs/1612.08242.
- [Redmon and Farhadi, 2017] Redmon, J. and Farhadi, A. (2017). Yolo9000: Better, faster, stronger. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6517–6525.
- [Ren et al., 2017] Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149.
- [Selimovi et al., 2018] Selimovi, A., Meden, B., Peer, P., and Hladnik, A. (2018). Analysis of content-aware image compression with vgg16. In *2018 IEEE International Work Conference on Bioinspired Intelligence (IWobi)*, pages 1–7.
- [Shi et al., 2015] Shi, B., Fan, T., and Liu, Q. (2015). Online object tracking and learning with sparse deformable template models. In *2015 8th International Symposium on Computational Intelligence and Design (ISCID)*, volume 2, pages 67–70.
- [Streit, 2016] Streit, R. (2016). Jpda intensity filter for tracking multiple extended objects in clutter. In *2016 19th International Conference on Information Fusion (FUSION)*, pages 1477–1484.
- [Wang et al., 2018] Wang, B. H., Wang, Y., Weinberger, K. Q., and Campbell, M. (2018). Deep person re-identification for probabilistic data association in multiple pedestrian tracking. *CoRR*, abs/1810.08565.
- [Wojke et al., 2017] Wojke, N., Bewley, A., and Paulus, D. (2017). Simple online and realtime tracking with a deep association metric. *CoRR*, abs/1703.07402.

[Xu et al., 2019] Xu, Y., Ban, Y., Alameda-Pineda, X., and Horaud, R. (2019). Deepmot: A differentiable framework for training multiple object trackers. *CoRR*, abs/1906.06618.

[Zhu et al., 2018] Zhu, Z., Wang, Q., Li, B., Wu, W., Yan, J., and Hu, W. (2018). Distractor-aware siamese networks for visual object tracking. *CoRR*, abs/1808.06048.