UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Posgrados

Estimación de Velocidad en Video Usando Técnicas de Visión Computacional y Redes Neuronales

Proyecto de Titulación

Erick Esteban Gallardo Ortiz

Israel Pineda, Ph.D.

Director de Trabajo de Titulación

Trabajo de titulación de posgrado presentado como requisito para la obtención del título de Magíster en Ciencia de Datos

Quito, 01 de Diciembre del 2024

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ
COLEGIO DE POSGRADOS

HOJA DE APROBACIÓN DE TRABAJO DE TITULACIÓN

Estimación de Velocidad en Video Usando Técnicas de Visión Computacional y Redes Neuronales

Erick Esteban Gallardo Ortiz

Nombre del Director del Programa:	Felipe Grijalva
Título académico:	Ph.D. en Ingeniería Eléctrica
Director del programa de:	Ciencia de Datos
Nombre del Decano del colegio Académico:	Eduardo Alba
Título académico:	Doctor en Ciencias Matemáticas
Decano del Colegio:	Ciencias e Ingenierías
Nombre del Decano del Colegio de Posgrados:	Dario Niebieskikwiat
Título académico:	Doctor en Física

Quito, Diciembre 2024

© DERECHOS DE AUTOR

Nombre del estudiante:	Erick Esteban Gallardo Ortiz
Código de estudiante:	00338630
C.I.:	1725087058
Lugar y fecha:	Quito, 01 de Diciembre de 2024.

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en http://bit.ly/COPETheses.

UNPUBLISHED DOCUMENT

Note: The following graduation project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on http://bit.ly/COPETheses.

DEDICATORIA

A mis padres, por su amor, paciencia y apoyo incondicional, quienes han sido mi mayor inspiración y fortaleza en este camino.

A mi familia, por estar siempre a mi lado, creyendo en mis sueños y alentándome a superarme cada día, en especialmente a mis primos Esteban y Milena Ortiz que han sido un pilar fundamental en mi vida.

A Madeleine Miranda, por su compañía, palabras de ánimo y por recordarme que todo esfuerzo tiene su recompensa y recordarme cada día que debo seguirme superando y ser mejor cada día.

AGRADECIMIENTOS

Agradezco a todas las personas y a la Universidad San Francisco que hicieron posible la realización de este trabajo.

En primer lugar, a mi Tutor Israel Pineda, por su guía y apoyo durante todo este proceso, por su conocimiento para cumplir con los objetivos planteados.

A mis profesores y compañeros de la Universidad San Francisco, con quienes he compartido tantas horas.

A mi familia, por se siempre un apoyo incondicional y por creer en mi los momentos mas difíciles.

A mi novia, que siempre esta conmigo dándome apoyo emocional y me ayuda a ser mejor cada día.

Finalmente, a mis amigos por estar siempre conmigo y por lo momentos compartidos, que hacen que cada día sea mas llevadero.

RESUMEN

Este trabajo propone la comparación entre modelos pre-entrenados y un modelo sin entrenamiento, todos estos basados en redes convolucionales con el fin de poder estimar la velocidad en base a videos en tiempo real. Todo esto utilizando un dataset en el cual se tiene etiquetado las velocidades en un instante de tiempo, los modelos tienen como arquitectura 3D CNN para capturar las características espaciales y temporales que ofrece la sucesión de imágenes. Los resultados muestran que los modelos pre-entrenados ofrecen una mejor adaptación al problema comparado con un modelo sin pesos inicializados, de acuerdo con estos resultados se observa que de manera general los modelos pre-entrenados convergen en menos épocas y menor tiempo de entrenamiento, a pesar de consumir más recursos computacionales.

Palabras clave: Estimación de velocidad de Vehículos, Visión por computadora, Redes Neuronales Convolucionales 3D, Modelos pre-entrenados, Aprendizaje profundo

ABSTRACT

This work proposes a comparison between pre-trained models and a model without prior training, all based on convolutional neural networks, with the aim of estimating speed from real-time videos. The dataset used contains labeled speeds at specific time instances, and the models are built with a 3D CNN architecture to capture the spatial and temporal features offered by the sequence of images. The results show that pre-trained models provide better adaptation to the problem compared to a model without initialized weights. In general, pre-trained models converge in fewer epochs and require less training time, although they consume more computational resources.

Key words: Vehicle Speed Estimation,Computer Vision, 3D Convolutional Neural Networks, Pre-trained Models, Deep Learning

TABLA DE CONTENIDO

Contents

ÍNDICE DE TABLAS

List of Tables

ÍNDICE DE FIGURAS

List of Figures

Estimación de Velocidad en Video Usando Técnicas de Visión Computacional y Redes Neuronales

Erick Gallardo, Israel Pineda

***Abstract*—Este trabajo propone la comparación entre modelos pre-entrenados y un modelo sin entrenamiento, todos estos basados en redes convolucionales con el fin de poder estimar la velocidad en base a videos en tiempo real. Todo esto utilizando un dataset en el cual se tiene etiquetado las velocidades en un instante de tiempo, los modelos tienen como arquitectura 3D CNN para capturar las características espaciales y temporales que ofrece la sucesión de imágenes. Los resultados muestran que los modelos pre-entrenados ofrecen una mejor adaptación al problema comparado con un modelo sin pesos inicializados, conforme con estos resultados se observa que de manera general los modelos pre-entrenados convergen en menos épocas y menor tiempo de entrenamiento, a pesar de consumir más recursos computacionales.**

***Palabras clave*—Estimación de velocidad de Vehículos, Visión por computadora, Redes Neuronales Convolucionales 3D, Modelos pre-entrenados, Aprendizaje profundo**

I. Introducción

La estimación de velocidad hoy en día es muy importante para el monitoreo y control de vías, no solo debido a que por el exceso de este puede ocasionar accidentes de tránsito, sino que en ciertos sectores se producen embotellamientos por lo que es importante llevar un control de la velocidad media para poder determinar en que sectores se debe mejorar la velocidad vial. Este documento presenta una investigación enfocada en la estimación de la velocidad utilizando metodologías del aprendizaje profundo.

La visión computacional ha demostrado ser capaz de resolver tareas de este tipo, en el cual se requiere extraer información espacial y temporal para obtener una estimación cercana a la realidad de la velocidad. Muchos de estos sistemas automatizados de reconocimiento de velocidad de vehículos dependen en gran medida de estimaciones precisas de distancias en rangos de tiempo. Aunque el mecanismo más utilizado históricamente es por medio de máquinas establecidas(cinemómetros).

A. Importancia de la Velocidad del Vehículo

En este contexto, la importancia de estimar con precisión la velocidad de los vehículos se vuelve aún más pronunciada. Las técnicas de aprendizaje profundo ofrecen un camino prometedor para mejorar la precisión y eficiencia de la estimación de la velocidad del vehículo[1], especialmente cuando se combinan con sistemas de detección y seguimiento[2].

B. Gestión de tráfico en tiempo real

La estimación precisa de la velocidad del vehículo es crucial para varias aplicaciones dentro de la gestión del tráfico, la planificación del transporte y la seguridad vial. En el ámbito de la gestión del tráfico, el conocimiento en tiempo real de las velocidades de los vehículos permite a las autoridades identificar puntos críticos de congestión, optimizar el flujo de tráfico e implementar estrategias de enrutamiento efectivas[3].

Al estimar con precisión las velocidades de los vehículos, las agencias de transporte pueden gestionar pro activamente los patrones de tráfico, reducir los tiempos de viaje y minimizar el impacto ambiental de las emisiones vehículares lo que permite la implementación de sistemas adaptativos para regular el tráfico, por medio de semáforos inteligentes[4].

C. Seguridad y concientización vial

La estimación precisa de la velocidad del vehículo es fundamental para garantizar la seguridad vial y mejorar la conciencia del conductor[5]. Al monitorizar y analizar las velocidades de los vehículos, los sistemas avanzados de asistencia al conductor pueden alertar a los conductores sobre posibles peligros, hacer cumplir los límites de velocidad y mitigar el riesgo de accidentes[6]. Además, la estimación precisa de la velocidad facilita el desarrollo de sistemas de transporte inteligentes capaces de adaptarse dinámicamente

a las condiciones cambiantes de la carretera y la dinámica del tráfico[7].

D. Planificación urbana

Analizar patrones de velocidad de diferentes zonas, ayuda a los planificadores urbanos a identificar zonas en las cuales se producen cuellos de botella en las infraestructuras viales, esto ayuda a realizar la toma de decisiones en base a datos con los cuales se puede elegir si implementar ampliaciones de vías o construir nuevas[8].

E. Abaratar costos

Estimar la velocidad de vehículos puede ser una tarea complicada, por lo que normalmente por medio de cinemómetros, sin embargo estos dispositivos poseen un costo elevado lo que limita su implementación en muchas áreas. Por lo que el uso de cámaras surge como una alternativa más accesible y económica, para reducir costos sin sacrificar la precisión[9].

En base a todo lo mencionado anteriormente se plantea la utilización de redes neuronales 3D como solución para la estimación de la velocidad a partir de videos previamente etiquetados, las redes neuronales son capaces de extraer características espaciales y temporales con las cuales pueden extraer patrones en cambios de imágenes secuenciales para identificar finalmente la velocidad del vehículo.

F. Retos Técnicos y Limitaciones

Uno de los principales desafíos al utilizar videos radica en la diversidad de entornos donde los vehículos operan. Esto incluye las condiciones climáticas (lluvias, niebla o nieve), iluminación y calidad de imagen e incluso la posición en la que se encuentra la cámara debido a que si se encuentra en una posición, todas estas condiciones juegan un papel crucial al momento de predecir la velocidad del vehículo, por ultimo los datos también son limitados, por lo que es un proceso costos y complejo debido a que tienen que ser etiquetados de manera manual[10].

II. Estado del arte

Esta sección se profundiza acerca de como ha ido evolucionando la estimación de velocidad, en primeras instancias utilizando hardware especializado, como cinemómetros y cámaras LIDAR[11], pasando a la utilización de algoritmos más avanzados de visión de computadora y aprendizaje profundo de maquina. Se presenta un análisis de trabajos más relevantes en el área, destacando las contribuciones realizadas y los enfoques de cada uno.

A. Hardware especializado

Tradicionalmente se han utilizado cinemómetros y cámaras LIDAR como métodos más utilizados para medir la velocidad de vehículos. Estos son dispositivos que actualmente siguen siendo utilizados en la actualidad, sin embargo como se explico anteriormente tiene estos tienen la desventaja de que poseen costos elevados debido a que se necesita técnicos especializados para la instalación y mantenimiento de los mismos, otra desventaja es que no se puede realizar un escalado masivo debido al costo de la implementación del mismo[11].

B. Visión computacional

Una de las técnicas más utilizadas en la actualidad, la visión computacional ha permitido el desarrollo de soluciones más económicas al aprovechar cámaras convencionales para la estimación de la velocidad por medio del seguimiento de los vehículos que aparecen en distintos tipos de entornos, esto utilizando marcas de detección con las cuales se define una velocidad máxima que se asume es constante para detectar posibles conductores que este fuera del rango[12][13].

Se han desarrollado varias tecnologías avanzadas para la visión de computación:

1) You Only Look Once (YOLO): Esta popular familia de modelos sobresale en la detección de objetos en tiempo real debido a su arquitectura de un solo paso. Predice eficientemente cuadros delimitadores y probabilidades de clase para objetos directamente a partir de características de imagen, lo que lo hace adecuado para tareas de procesamiento de vídeo[14].
2) Simple Online and Realtime Tracking (SORT): SORT es un método simple para realizar seguimientos de objetos en tiempo real. Utiliza algoritmos como el filtro de

Kalman extendido (EKF) o el filtro de Kalman unscented (UKF) para estimar la trayectoria de los objetos en movimiento. Este enfoque lo hace adecuado para aplicaciones de seguimiento en vídeo vigilancia y sistemas de monitoreo [15].

3) Single Shot MultiBox Detector (SSD): Similar a YOLO, SSD es un detector de un solo paso que prioriza la velocidad y la eficiencia, lo que lo hace adecuado para aplicaciones en tiempo real. Su enfoque en equilibrar precisión y velocidad lo convierte en otra opción viable para la detección de vehículos en la vigilancia del tráfico.[16]
4) Filtro de Kalman: El Filtro de Kalman es una técnica ampliamente utilizada en el campo del seguimiento de objetos. Se basa en un modelo de espacio de estados para predecir las posiciones futuras de los objetos en función de sus estados anteriores y de las mediciones actuales. Este filtro es conocido por su eficiencia y robustez contra el ruido en los datos. Sin embargo, puede enfrentar dificultades cuando hay cambios repentinos en el movimiento del objeto, ya que su modelo lineal asume un movimiento suave y predecible[17].
5) Flujo Óptico: El Flujo Óptico es una técnica que estima el movimiento de los objetos analizando el desplazamiento aparente de los píxeles entre fotogramas consecutivos de una secuencia de vídeo. Esta técnica es computacionalmente eficiente y puede proporcionar información detallada sobre el movimiento de los objetos en la escena. Sin embargo, puede ser sensible a cambios en la iluminación y a oclusiones, lo que puede afectar su precisión en ciertas condiciones[18].

C. Aprendizaje Profundo

La aplicación de algoritmos de aprendizaje profundo se esta volviendo cada vez más común, por lo que se han realizado experimentos que combinan la detección y seguimiento de vehículos en conjunto con redes neuronales. Estos utiliza la detección y seguimiento para medir el desplazamiento del vehículo, mientras que las redes neuronales analizan las características de dicha secuencia para poder predecir la velocidad[19]

III. Materiales y Metodología

En la figura 1 se presenta el flujo general empleado en este caso de estudio, destacando las principales etapas para la estimación de la velocidad de vehículos.

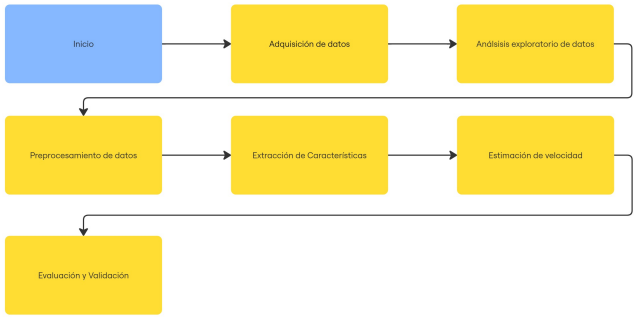

Figure 1. Pipeline utilizado en el modelo

A. Adquisición de datos

En este trabajo se emplearon cámaras, las cuales captan varios tipos de vehículos a velocidad constante, en una carretera. La cámara está colocada de manera fija, este dataset posee 13 tipos de vehículos distintos variando entre: marca, año de producción, tipo de motor, potencia y transmisión, lo que hace que el dataset posea 400 grabaciones de video en una calidad de 1920 x 1080 pixeles a 30 cuadros por segundo[20].

B. Análisis exploratorio de los datos

Se comenzó realizando el análisis exploratorio de todas las velocidades registradas en los distintos tipos de vehículos, en los cuales notamos que existe una distribución normal de los datos, en donde los valores menos probables se encuentran en las colas de la misma. Esto sugiere que para velocidades superiores a 100 km/h y por debajo de 40 km/h existe una carencia de datos para poder generalizar de manera efectiva las velocidades esto se observa en la figura 2, lo que podría limitar la capacidad del modelo para realizar estimaciones precisas en dichos rangos.

Posteriormente se realizó el análisis por vehículo como se muestra en la figura 3 en el cual se observan los rangos interquartiles de las velocidades por cada vehículo, en el cual es importante resaltar que todos los vehículos se encuentran en los mismos rangos de velocidad, con lo que se puede afirmar

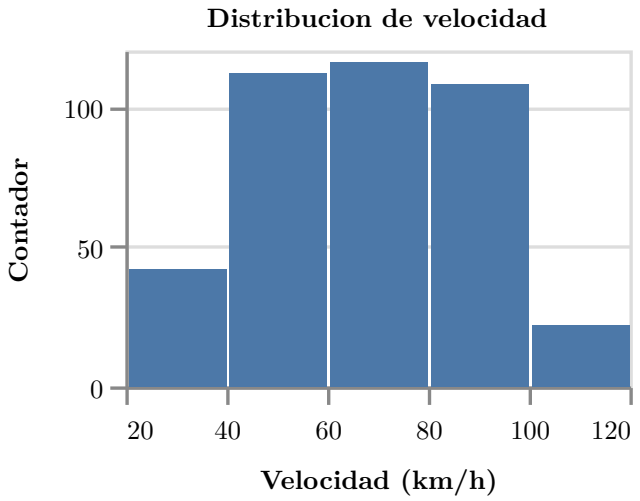

Figure 2. Distribución de velocidades de vehículos

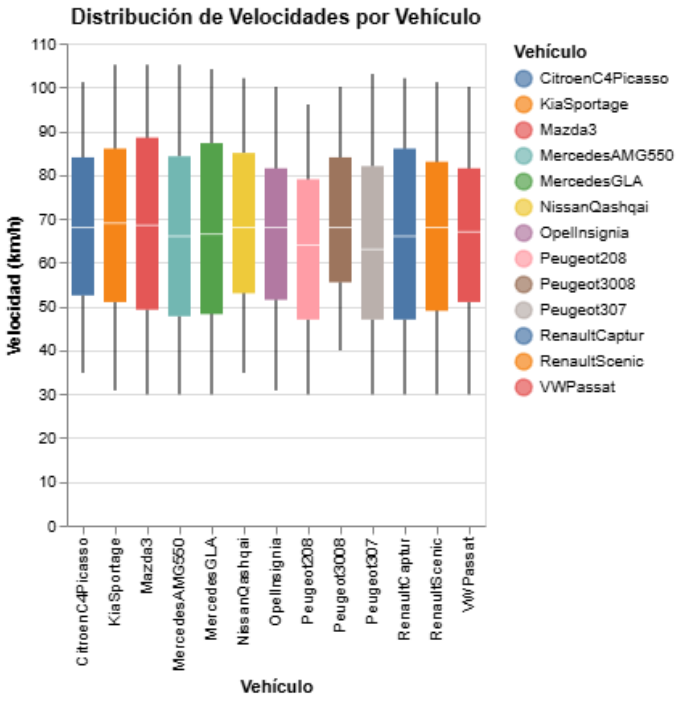

Figure 3. Distribución de Velocidades por Vehículo

que los datos son uniformes en el rango de 45 km/h a 100 km/h. Sin embargo, se puede notar que existen algunos modelos como el Mazda 3 y el Mercedes GLA, los cuales tienden a tener una mayor dispersión, esto evidenciado con el tamaño de las cajas de los mismos. Por otro lado tenemos el caso contrario modelos como el peugeot3008 y el VWPassat muestran una menor dispersión por lo que, se evidencia en un gráfico más compacto que los anteriores mencionados.

Es importante recalcar que al igual que en la figura 2 se observa que los valores menores a 40 km/h y superiores a los 100 km/h se consideran valores atípicos, por lo que se puede afirmar nuevamente que para futuras investigaciones se necesitaría tomar más datos en dichos rangos.

Continuando con el análisis de los datos, se realizó un análisis con respecto al tiempo, tomando en cuenta que en el dataset no tiene la velocidad a cada instante del vehículo es importante realizar un análisis para explorar una posible correlación entre velocidad y tiempo, como se muestra en la figura 4 se observa que no existe una correlación entre la velocidad y el tiempo, pero este gráfico también brinda información importante acerca de los valores mínimos y máximos de tiempo registrado en los videos. Se puede observar que el tiempo mínimo y máximo son aproximadamente a los 4.5 segundos y 7.5 segundos respectivamente. Esto nos permite inferir que en videos de corta duración la mayor parte del tiempo el vehículo no sera visualizado en el video. Por otro lado en los videos de mayor duración es probable que el vehículo no salga en las primeras instancias del video. Esta información es importante debido a que los modelos de aprendizaje profundo suelen utilizar muchos recursos computacionales.

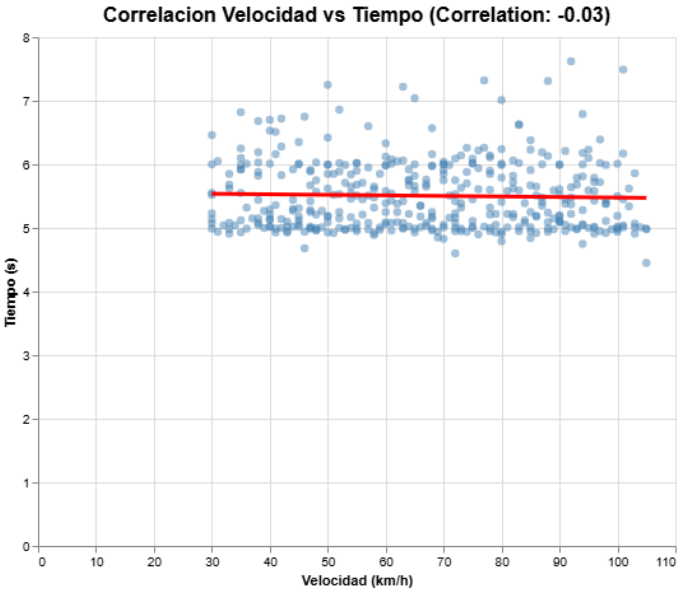

Figure 4. Correlación de tiempo y velocidad

C. Materiales

Para la implementación de este trabajo se utilizaron diversos recursos de hardware, software y los videos anteriormente mencionados, con estos insumos se realizó la construcción de un pipeline para la estimación de la velocidad de vehículos utilizando videos.

1) Hardware: En cuanto a hardware se utilizó una computadora con una GPU dedicada para el entrenamiento de modelos de aprendizaje profundo, en especifico un tarjeta RTX 4060 Ti de 16 GB, en cuanto al almacenamiento de la Ram, se utilizó 32 GB todo esto es necesario debido a que como se explico anteriormente son 400 videos.
2) Software: Se utilizó varias librerías enfocadas en tareas de visión computacional y aprendizaje profundo. Se han utilizado herramientas como pytorch para facilitar la construcción y gestión de las redes neuronales. Para realizar la carga de los videos en formato MP4 se utilizó moviepy, para posteriormente realizar el preprocesamiento de los videos utilizado opencv aplicando técnicas de redimensionamiento en cada recuadro de los videos.

D. Experimentos

E. Preprocesamiento de datos

Se realizaron diversos experimentos para la optimización del uso de recursos y garantizar el procesamiento de los 400 videos disponibles en el conjunto de datos. No se puede realizar la carga de todos los videos debido a que excede la capacidad física del ordenador, se realizó un proceso de variación de distintos parámetros enfocado en utilizar la mayor cantidad de datos con el hardware disponible. Durante estos experimentos, se determinó que el número máximo de cuadros manejables manteniendo una resolución de 160 x 160 x 3 pixeles es de 120 cuadros por video, con esta configuración se logro optimizar los recursos disponibles, hasta alcanzar un total de 31.2GB de 32GB disponibles. Este ajuste garantiza la viabilidad del procesamiento de datos sin sacrificar la calidad de los datos utilizados.

F. Extracción de características

Se realizó el entrenamiento con estos modelos variando hiperparametros clave, como la tasa del aprendizaje y varios optimizadores. Se realizó la evaluación con diversos optimizadores entre los cuales se encuentran adam, adam con decaimiento, adagrad, RMSprop y SGD. Los resultados muestran que los valores más óptimos a utilizar es el optimizador adam con decaimiento de la pérdida de 0.0001 y una tasa de aprendizaje de 0.00003, debido a que este produjo la menor pérdida durante la etapa de validación.

Para la implementación de los modelos de aprendizaje profundo se tomo en consideración que la RTX 4060 Ti tiene 16 Gb de ram como se menciono antes, por lo que el numero máximo de 6 videos por lote de entrada para el modelo sin pesos, estos ocupaban 13Gb de los 16Gb disponibles en la tarjeta gráfica, se utilizó una arquitectura basada en experimentación la cual se basa en una serie de capas en la cual se utiliza una convolución, seguido de una normalización por lotes y por ultimo una capa de agrupamiento máximo, este patrón se repite seis veces, con la función de activación LeakyReLU, ya que demostró ofrecer una mejor adaptación que ReLu. Después de estas capas de convolución y agrupación se realizó un aplanamiento para convertir las salidas tridimensionales a un vector unidimensional, finalmente se aplican dos redes totalmente conectadas con función de activación ReLu debido a que posee un comportamiento en el rango de 0 y positivos, alineándose con el dominio de la velocidad.

Posteriormente, para la selección de modelos preentrenados, se tomo en consideración los modelos disponibles en el framework de pytorch, en concreto en la versión 2.0 que es la más estable actualmente, en esta versión se encuentran disponibles tres modelos (MC3_18, ResNet3D y r2plus1d_18) los cuales utilizan millones de parámetros[21]. Sin embargo, solo se utilizó dos de ellos, esto se debe a que, al utilizar r2plus1d_18, el consumo de memoria superaba el total de los recursos computacionales disponibles, es importante recalcar que de igual manera con los modelos MC3_18 y ResNet3D el máximo tamaño por lote era 1 evidenciando así que a pesar de tener la capacidad de generalizar mejor y converger más rápido, estos

consumen más recursos, por lo que puede existir una limitación en el hardware.

Posteriormente se realizó una división en el conjunto de datos en tres partes: un 60% utilizado para entrenamiento del modelo, 20% utilizado para validación del modelo y 20% utilizado para realizar pruebas con datos independientes del proceso de entrenamiento, se decidió realizar así la división debido a que como se menciona en [22] la manera más efectiva de dividir un dataset es en 70% para entrenamiento y 30% para pruebas. Sin embargo, se decidió tomar un 10% de cada uno de estos conjuntos para formar un conjunto de validación, lo que permite realizar una evaluación objetiva del modelo durante la fase de resultados.

Se realizó la experimentación con un numero máximo de 120 épocas, esto debido a que el conjunto de entrenamiento fue de 300 videos, normalmente se debería considerar un máximo de 50 épocas para pasar por todos los videos, pero en este caso se decidió incrementar el numero de épocas debido a la tasa de aprendizaje tan baja. Además, se implementó un mecanismo de parada temprana, que interrumpe el entrenamiento cuando el modelo no ha sido capaz de mejorar en 10 épocas seguidas, evitando el desperdicio de tiempo y recursos computacionales.

Estos hallazgos resaltan la importancia de ajustar los datos para su procesamiento, realizar la variación de hiperparametros para obtener un rendimiento optimo en el modelo. Además, se demostró que en general los modelos pre-entrenados tienen una ventaja sobre el modelo sin pesos inicializados, esto debido a que estos convergen en menos épocas, a pesar de utilizar más memoria en la tarjeta de video.

IV. Resultados y Discusión

A. Estimación de velocidad

Realizando el entrenamiento con tomando los hiperparametros más óptimos, es decir una tasa de aprendizaje de 0.00003 y el optimizador adam con decaimiento de la pérdida de 0.0001 se obtienen las figuras 5 y 6 en las cuales se observa que el modelo que converge en menos épocas es el ResNet3D, mientras que una arquitectura sin un entrenamiento previo demora muchas más épocas y como se muestra en la tabla I el que alcanza

Table I
Comparación de MSE y épocas de los modelos

Modelo	MSE mínimo	época
CNN3D	14.560210592606488	106
MC3_18	11.983617936863618	85
ResNet3D	16.90047182581004	74

el mínimo valor es MC3_18 por lo que es el seleccionado para realizar las predicciones sobre el conjunto de pruebas.

Con base en todo lo realizado, se llevó acabo la estimación de velocidad sobre el conjunto de videos que no formaron parte del entrenamiento ni de la validación. Esto con el fin de garantizar que la estimación se realizara sobre datos no sesgados para obtener medidas objetivas del rendimiento del mejor modelo.

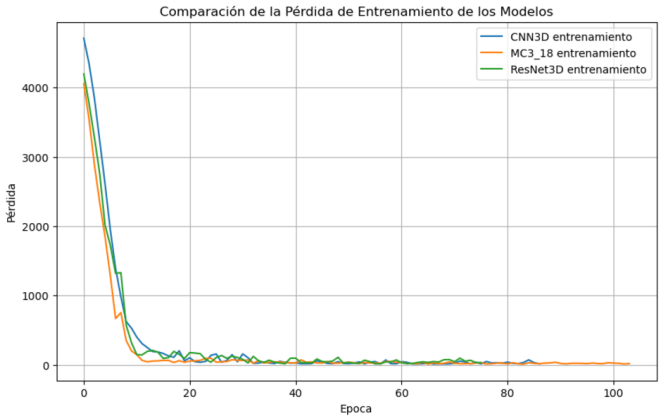

Figure 5. Comparación de la Pérdida de Entrenamiento de los Modelos

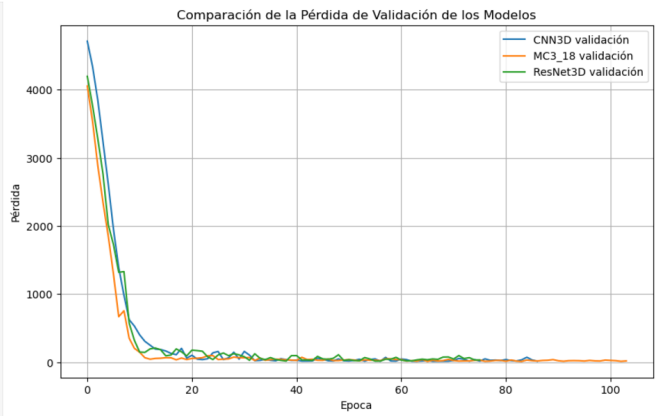

Figure 6. Comparación de la Pérdida de Validación de los Modelos

B. Evaluación y validación

Se realizó la evaluación en una muestra de 57 videos como se muestra en la imagen 7 las velocidades predichas se adaptan a las velocidades

reales adaptándose de manera consistente con las predicciones, lo que indica que el modelo mantiene una buena capacidad de generalización. Sin embargo, como se menciono antes los valores menores a 40 son difíciles de predecir, lo que sugiere que se debería aumentar muestras con vehículos a baja velocidad, en este caso no se tomo velocidades mayores a 100. Sin embargo se recomienda aumentar dichos datos debido a que no se tiene una gran cantidad de los mismos, solo valores próximos.

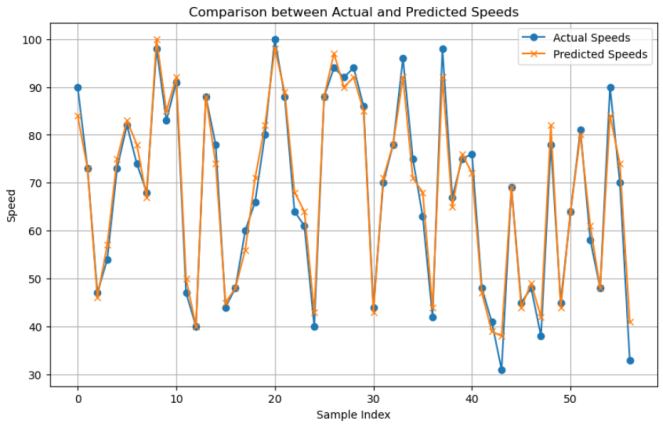

Figure 7. Comparación entre velocidades

En comparación con trabajos relacionados como [19] se observa que el error medio absoluto es aproximadamente 3.02 km/h y en error medio cuadrático 17.35 km/h, mientras que el modelo implementado tiene un error medio absoluto de 2.8 km/h y en error medio cuadrático 3.59 km/h por lo que se puede decir que ofrece una mejor estimación de las velocidades. Sin embargo este modelo presenta un limitaciones, como la dependencia de una posición especifica de la cámara y la incapacidad para manejar múltiples vehículos. Además de que en los datos utilizados solo se tiene condiciones de visibilidad perfecta, por lo que restringe su aplicabilidad en el mundo real. Por lo que para implementar un modelo que sea capaz de adaptarse a la complejidad del mundo real, se debería implementar más datos, tales como vehículos a mayor velocidad, menor velocidad, condiciones de baja visibilidad, diferentes condiciones climáticas, distintos escenarios, distintos ángulos y distinta posición de la cámara y por ultimo más vehículos en un mismo video, lo cual se puede comprobar en el repositorio https://github.com/erick2611gaeg/SpeedEstimation.

V. Conclusiones

1) Desafíos de iluminación y ambiente: Los diferentes entornos pueden presentar condiciones variables de iluminación, como luz diurna brillante, luz baja o incluso condiciones de poca luz o nocturnas lo que dificulta la visión por computación, también puede verse afectada por factores como la lluvia, la niebla o la nieve pueden afectar la visibilidad y la apariencia de los automóviles. Los sistemas de detección deben ser capaces de adaptarse y mantener un rendimiento consistente bajo diversas condiciones climáticas. Por ultimo los automóviles pueden encontrarse en una variedad de entornos, desde áreas urbanas hasta entornos rurales. Los algoritmos de detección deben ser capaces de distinguir los automóviles del fondo y adaptarse a diferentes paisajes y contextos
2) Importancia de la precisión: La detección de automóviles es crucial para aplicaciones como la conducción autónoma, la gestión del tráfico y la seguridad vial. Por lo tanto, es fundamental que los sistemas de detección logren altos niveles de precisión y fiabilidad para garantizar un funcionamiento seguro y eficiente.
3) Avances en tecnologías de visión por computadora: Los avances en técnicas de visión por computadora, como el aprendizaje profundo y la inteligencia artificial, han mejorado significativamente la capacidad de detectar automóviles en diferentes entornos. Estas tecnologías permiten el desarrollo de sistemas más robustos y precisos que pueden adaptarse a una amplia gama de condiciones.
4) Eficiencia de modelos pre-entrenados: Los modelos pre-entrenados demostraron ser significativamente más eficientes en términos de tiempo y eficiencia comparado con un modelo sin pesos inicializados.
5) Condiciones ideales vs mundo real: El modelo demostró buenos resultados en condiciones ideales, esto limita su aplicabilidad en entornos en los cuales las condiciones son muy variantes y que pueden afectar a las predicciones del mismo.
6) Limitación de datos: A pesar de los buenos resultados obtenidos, se identifico que se de-

bería aumentar datos en velocidades menores a 40km/h y mayores a 100km/h para aumentar la generalización dentro de estas condiciones especificas.

References

[1] G. Wang, J. Li, P. Zhang, X. Zhang, and H. Song, "Pedestrian speed estimation based on direct linear transformation calibration," in *2014 International Conference on Audio, Language and Image Processing*. IEEE, 2014, pp. 195–199.

[2] J. Ma, D. Li, L. Li, S. Xue, Y. Wang, and J. Jia, "Multi-target tracking control system of phased array antenna with mechanical scanning and beam scanning method," in *2023 2nd Conference on Fully Actuated System Theory and Applications (CFASTA)*. IEEE, 2023, pp. 216–221.

[3] S. Gupta and B. Sundar, "A computer vision based approach for automated traffic management as a smart city solution," in *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. IEEE, 2020, pp. 1–6.

[4] S. S. Chavan, R. Deshpande, and J. Rana, "Design of intelligent traffic light controller using embedded system," in *2009 Second International Conference on Emerging Trends in Engineering & Technology*. IEEE, 2009, pp. 1086–1091.

[5] G. G. L. Cruz, A. Litonjua, A. N. P. San Juan, N. J. Libatique, M. I. L. Tan, and J. L. E. Honrado, "Motorcycle and vehicle detection for applications in road safety and traffic monitoring systems," in *2022 IEEE Global Humanitarian Technology Conference (GHTC)*. IEEE, 2022, pp. 102–105.

[6] H. Chen, F. Zhao, K. Huang, and Y. Tian, "Driver behavior analysis for advanced driver assistance system," in *2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS)*. IEEE, 2018, pp. 492–497.

[7] F.-Y. Wang, Y. Lin, P. A. Ioannou, L. Vlacic, X. Liu, A. Eskandarian, Y. Lv, X. Na, D. Cebon, J. Ma *et al.*, "Transportation 5.0: The dao to safe, secure, and sustainable intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[8] M. T. Gómez, "La ciudad, para quién: desafíos de la movilidad a la planificación urbana," *Biblio 3w: revista bibliográfica de geografía y ciencias sociales*, 2018.

[9] J. M. Huidobro, "Radares para el control del tráfico," *ACTA, Madrid*, 2016.

[10] S. Dodge and L. Karam, "Understanding how image quality affects deep neural networks," in *2016 eighth international conference on quality of multimedia experience (QoMEX)*. IEEE, 2016, pp. 1–6.

[11] F. Zhang, D. Clarke, and A. Knoll, "Vehicle detection based on lidar and camera fusion," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2014, pp. 1620–1625.

[12] S. Hua, M. Kapoor, and D. C. Anastasiu, "Vehicle tracking and speed estimation from traffic videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 153–160.

[13] H. Rodríguez-Rangel, L. A. Morales-Rosales, R. Imperial-Rojo, M. A. Roman-Garay, G. E. Peralta-Peñuñuri, and M. Lobato-Báez, "Analysis of statistical and artificial intelligence algorithms for real-time speed estimation based on vehicle detection with yolo," *Applied Sciences*, vol. 12, no. 6, p. 2907, 2022.

[14] M. Mostafa, S. Sadi, S. A. Anamika, M. S. Hussain, and R. Khan, "Automatic vehicle classification and speed tracking," in *2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, 2023, pp. 972–977.

[15] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE international conference on image processing (ICIP)*. IEEE, 2017, pp. 3645–3649.

[16] V. Thakar, H. Saini, W. Ahmed, M. M. Soltani, A. Aly, and J. Y. Yu, "Efficient single-shot multibox detector for construction site monitoring," in *2018 IEEE International Smart Cities Conference (ISC2)*. IEEE, 2018, pp. 1–6.

[17] Q. Li, R. Li, K. Ji, and W. Dai, "Kalman filter and its application," in *2015 8th international conference on intelligent networks and intelligent systems (ICINIS)*. IEEE, 2015, pp. 74–77.

[18] Y. Li, F. Chen, F. Yang, C. Ma, Y. Li, H. Jia, and X. Xie, "Optical flow-guided mask generation network for video segmentation," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2020, pp. 1–5.

[19] H. Dong, M. Wen, and Z. Yang, "Vehicle speed estimation based on 3d convnets and non-local blocks," *Future Internet*, vol. 11, no. 6, p. 123, 2019.

[20] S. Djukanović, N. Bulatović, and I. Čavor, "A dataset for audio-video based vehicle speed estimation," in *2022 30th Telecommunications Forum (TELFOR)*. IEEE, 2022, pp. 1–4.

[21] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 6450–6459.

[22] Q. H. Nguyen, H.-B. Ly, L. S. Ho, N. Al-Ansari, H. V. Le, V. Q. Tran, I. Prakash, and B. T. Pham, "Influence of data splitting on performance of machine learning models in prediction of shear strength of soil," *Mathematical Problems in Engineering*, vol. 2021, no. 1, p. 4832864, 2021.