

# Estudio de factibilidad del uso de modelos de redes neuronales artificiales en la automatización del aforo y clasificación vehicular del transporte público



*Feasibility study of the use of artificial neural network models in the automation of vehicle count and classification of public transport*

Patricio Xavier Moreno Vallejo.<sup>1</sup>, Gisel Katerine Bastidas Guacho.<sup>2</sup> & Patricio René Moreno Costales.<sup>3</sup>

Recibido: 24-05-2020 / Aceptado: 13-06-2020 / Publicado: 03-07-2020

**Abstract.**

**DOI:** <https://doi.org/10.33262/concienciadigital.v3i3.1355>

The analysis of vehicular traffic has become important in recent years due to the considerable increase of vehicles on the roads, which often is translated in traffic congestion. So, it is necessary to know the vehicle capacity of the roads, in order to take decisions to improve road traffic such as opening alternate roads, changing the timing of traffic lights, improving road signs, and so on. There are studies that perform the counting and identification of vehicles manually by observation, however, this technique can be inefficient since it must have people who perform this count on one or more roads during a certain period of time, leaving out information of several hours of the day. On the other hand, there are techniques that can be complex to configure or that interfere in the normal flow of traffic such as ramps with sensors and inductive loops. Therefore, there is a need to use a technique that allows the counting and identification of vehicles in a more efficient way without interfering in the flow of traffic. This study proposes the use of artificial neural network models in the automation of vehicle counting and classification. An artificial neural network model was generated that, through video, allows the detection and classification

---

<sup>1</sup> Escuela Superior Politécnica de Chimborazo, Facultad de Administración de Empresas, Riobamba, Ecuador. xavier.moreno@esPOCH.edu.ec

<sup>2</sup> Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Riobamba, Ecuador. gis.bastidas@esPOCH.edu.ec

<sup>3</sup> Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Riobamba, Ecuador. pmoreno@esPOCH.edu.ec

of public transport vehicles that travel on the roads with an accuracy of 94.29%. In addition, the model was tested in real time using an app and a smartphone camera, demonstrating that the model can be used if video is available without the need for calibration or system configuration. Therefore, based on the high level of certainty obtained and the tests carried out in real time, the evidence demonstrate that it is feasible to create systems that do not require calibration or complex configurations based on the use of artificial neural networks for the automation of counting and classifying vehicles.

**Keywords:** Vehicular count, ANN, inference, classification.

### **Resumen.**

El análisis del tránsito vehicular ha tomado gran importancia en los últimos años debido al incremento considerable de vehículos en las vías, lo que hace que muchas veces exista congestiones vehiculares, por lo que, es necesario conocer el aforo vehicular que transita por determinadas vías, de tal forma que se puede tomar decisiones para mejorar el tránsito vial como abrir vías alternas, cambiar temporización de semáforos, mejorar señalética de las vías, etc. Existen estudios que realizan el conteo e identificación de vehículos de forma manual mediante observación, sin embargo, esta técnica puede ser ineficiente ya que se debe contar con personas que realicen este conteo en una o varias vías durante un periodo de tiempo determinado, además, no permite capturar la información en varias horas del día. Por otro lado, existen técnicas que pueden ser complejas de configurar o que intervienen en el flujo normal del tráfico como rampas con sensores y lazo inductivo. Por lo que se ve la necesidad de usar una técnica que permita el conteo e identificación de vehículos de una manera más eficiente y que no intervenga en el flujo del tránsito. El presente estudio plantea la utilización de modelos de redes neuronales artificiales en la automatización del aforo y clasificación vehicular. Se generó un modelo de redes neuronales artificiales que a través de video permite detectar y clasificar vehículos del transporte público que transitan por las vías con un nivel de certeza del 94.29%. Además, se probó el modelo en tiempo real mediante el uso de una aplicación móvil y la cámara del smartphone, demostrando que el modelo puede ser usado siempre que se tenga acceso a video sin la necesidad de una calibración o configuración del sistema. Por lo tanto, en base al alto nivel de certeza obtenido y las pruebas realizadas en tiempo real, queda demostrado que es factible la creación de sistemas que no requieran calibración ni configuraciones complejas basados en el uso de redes neuronales artificiales para la automatización del aforo y clasificación vehicular.

**Palabras claves:** aforo, vehicular, redes neuronales, inferencia, clasificación.

### **Introducción.**

Una de las ramas de la ingeniería de transporte es la ingeniería de tránsito la cual se encarga de estudiar todo lo correspondiente a las vías que forman parte de las redes viales primarias, secundarias y terciarias de un sector. En esta rama de la ingeniería se busca facilitar la toma de

decisiones en base a simulaciones y estudios de capacidad de vía para mejorar el tránsito vehicular. Sin embargo, para realizar dichos estudios es necesario poseer datos sobre el uso de la vía, los cuales a menudo son difíciles de obtener debido a que el número de vehículos que transitan por una vía no es un valor constante, sino que es un valor variable de naturaleza aleatoria. Lo mismo sucede con las características de los vehículos ya que son variables, por ejemplo, el color, la forma, el peso, la longitud. Por lo tanto, la tarea de obtener datos referentes a los tipos de vehículos (clasificación vehicular) y la cantidad de vehículos que transitan por una vía (aforo vehicular), muchas de las veces se la deben realizar de forma manual usando técnicas como la observación directa o a través de video. Esta técnica tiende a ser costosa debido a que se debe contratar personal que realice la observación y conteo correspondiente en una o varias vías durante un periodo de tiempo determinado. También tiende a ser imprecisa debido a que, por el error de muestreo, puede darse el caso de que el día que se realiza el aforo exista alguna eventualidad que provoque un incremento o decremento anómalo del tránsito vehicular y esto ocasionaría errores en los estudios relacionados con la capacidad de la vía. De igual forma, puede existir el error humano que por diferentes factores, como bloqueos visuales, pueden provocar errores en los valores reportados. (Minge, Petersen, y Kotzenmacher 2011) manifiestan que a menudo los errores de conteos manuales son pequeños y pueden ser ignorados, sin embargo, (Zheng y Mike 2012) indican que el nivel de certeza en los conteos manuales no han sido bien investigados y reportados; también indican que el impacto de los errores depende mucho de la aplicación y son menos del 1% cuando son tareas de conteo y entre el 4 al 5% cuando son tareas de clasificación. (Sharma 1983) realiza un estudio para analizar el nivel de certeza de los conteos manuales llegando a la conclusión que la precisión de los conteos cambia basado en el tipo de vía que van desde vías con un flujo bajo de vehículos hasta vías que tienen un alto flujo de vehículos; las variaciones de tráfico hora a hora durante el mismo día es un factor que también influye por lo cual al momento de realizar un conteo se debe seleccionar el horario más eficiente.

En el transcurso del tiempo se han llegado a definir varias metodologías para realizar el aforo vehicular. (Slinn 2006) describe tres de estas que son no intervencionistas, es decir, que no afectan el flujo de tráfico que está siendo medido: Conteos de tráfico automáticos, conteos manuales y movimientos de giro. Los conteos de tráfico automáticos a menudo se usan cuando se requiere obtener datos durante un periodo extenso y se lo hace mediante la utilización de tubos de presión o lazos inductivos que se encuentran fijos en algún punto de la vía. Por otra parte, los conteos manuales se emplean cuando no existe un mecanismo de conteo automático o se requiere recolectar datos en un periodo de tiempo corto. Por último, los conteos de movimiento de giro corresponden a la recopilación de datos de vehículos que pasan por una intersección; estos datos se diferencian de las otras metodologías debido a que intervienen otros factores como la aceleración y frenado de los diferentes tipos de vehículos que son variables al momento de aproximarse a una intersección.

Algunas formas de automatización del proceso de aforo vehiculares, incluye la propuesta realizada por (MONETTI et al. 2018), en donde a través de una aplicación móvil se recopilan los

datos, sin embargo, los datos deben ser ingresados manualmente por una persona por lo cual no varía mucho a lo que sería un conteo manual. (Alpízar y del Risco Sánchez 2012) proponen un sensor de flujo vehicular basado en lazo inductivo, sin embargo, el dispositivo debe ser instalado y fijado en un lugar debajo de la vía lo cual no da flexibilidad. (BETANCOURT 2019) propone una plataforma portátil para el aforo vehicular que consta de una rampa de ingreso, sensores de ingreso, plataforma de pesaje y rampa de salida con una longitud de 9 metros, sin embargo, esta propuesta se va en contra del principio de las metodologías de aforo vehicular que indica que las técnicas utilizadas no deben ser intervencionistas, y esta propuesta intervendría en el flujo normal de tránsito al obligar a los vehículos a reducir la velocidad para acceder y abandonar la plataforma mediante las rampas.

Una alternativa para automatizar el conteo vehicular sin intervenir en el flujo vehicular es utilizar cámaras de video, esta es una alternativa factible debido a que en los últimos años varias ciudades alrededor del mundo ya disponen de circuitos cerrados de televisión (CCTV) para el monitoreo del tráfico vehicular. China posee las ciudades con mayor número de cámaras en sus CCTV, sin embargo, ciudades de países que no son del primer mundo también poseen CCTV, incluso en algunas zonas de las ciudades principales de los países como Quito, Ecuador existen cámaras que forman parte de estos CCTV en cada intersección semaforizada. Sin embargo, muchas veces a pesar de que existe la tecnología que recopila miles de datos, en algunos casos durante las 24 horas del día, estos datos no son pre-procesados o explotados al máximo, por lo general estos datos son únicamente almacenados y nunca más se vuelven a visualizar.

Por otra parte, existen empresas como Iteris que ofertan sistemas complejos para el conteo de vehículos a través de videos, sin embargo, requieren de varios dispositivos electrónicos y un especialista que realice la calibración de los mismos. (Chintalacheruvu y Muthukumar 2012) proponen una alternativa basada en el algoritmo de detección de esquinas de Harris-Stephen (Harris y Stephens 2013); esta propuesta también requiere de una calibración inicial, posteriormente, recibe las capturas de video en tiempo real, pre procesa los fotogramas de video en base a la calibración inicial, se elimina el ruido de las imágenes debido a la baja resolución de las cámaras (320x240), convierte las imágenes de colores a escala de grises y aplica el algoritmo de Harris-Stephen para detectar los vehículos. (Avery, Wang, y Rutherford 2004) proponen un novedoso sistema de conteo de vehículos en base al trazado de una línea digital de la cual se toman los valores de los píxeles que la conforman, después analiza cada fotograma de video y compara si los valores de los píxeles de la línea digital han cambiado, una limitante es que se debe configurar un umbral de tal forma que si la diferencia absoluta de los valores RGB de al menos 40% de los píxeles de la línea es mayor a dicho umbral entonces se contabiliza un nuevo vehículo.

En este artículo se pretende determinar si es factible aplicar nuevas técnicas en la automatización del conteo de vehículos como son modelos de Deep Learning (Aprendizaje Profundo) que incluyen redes neuronales artificiales (ANN, por su siglas en inglés). La ventaja de esta propuesta

es que no necesita calibración de ningún tipo, ni configuraciones por parte del usuario, simplemente se necesita tener acceso al video captado por algún CCTV dado que como el modelo de predicción es entrenado previamente detectará los objetos indicados, en este caso vehículos. Algunas de las limitaciones de las redes neuronales artificiales, como mencionan (Mishra y Srivastava 2014), es que no permiten describir como se soluciona el problema o, en otras palabras, funcionan como una caja negra y provocan una gran carga computacional.

El artículo está organizado de la siguiente manera: en la sección Metodología se describe la metodología utilizada para la creación del modelo de Deep Learning que permite detectar vehículos de transporte público como buses y taxis. En la sección Resultados se presenta la aplicación del modelo en un conjunto de datos de prueba y se obtiene el nivel de certeza del mismo. En la sección Discusión se revisan artículos similares y se comparan los resultados obtenidos, adicionalmente, se presentan trabajos futuros que se pueden derivar de esta investigación. Por último, en la sección Conclusiones se establecen las conclusiones a las que se ha llegado con la presente investigación.

### **Metodología.**

El objetivo del presente estudio es obtener un modelo funcional de redes neuronales artificiales que permita detectar en un video la presencia de un vehículo y clasificar el vehículo según su tipo: Bus, Taxi u Otro. Para cumplir con dicho objetivo, primeramente, se obtuvo una muestra de imágenes de vehículos que transitaban por distintas vías de la ciudad de Riobamba. Las imágenes fueron tomadas desde diferentes ángulos de tal forma que el modelo pueda aprender a clasificar los vehículos desde distintos ángulos, de esta forma no será necesario colocar las cámaras en un ángulo específico. Posteriormente, las imágenes fueron recortadas con la finalidad de tener un solo vehículo por imagen para que pueda ser procesado y aprendido por el modelo. Luego las imágenes fueron agrupadas en tres categorías: Bus, Taxi, y Otro. A cada categoría se la agrupó en un conjunto de datos de entrenamiento y uno de pruebas. El conjunto de datos entrenamiento se utilizó para entrenar al modelo, es decir, para que el modelo aprenda a identificar y clasificar los vehículos. El segundo conjunto de datos de pruebas, en cambio, se utilizó para medir el nivel de certeza del modelo con datos que no han sido vistos previamente.

Después de obtener los datos, se procedió a crear un proyecto de Python para el desarrollo del modelo. Para el mismo se utilizaron librerías de código abierto como son: Pandas para la manipulación de datos, numpy para el soporte de arreglos multidimensionales, Tensorflow para machine learning, y Keras para Deep Learning. Con las librerías importadas, se procedió a cargar los conjuntos de datos de entrenamiento y pruebas en Python redimensionando todas las imágenes a la misma resolución de 256x256 pixeles debido a que es necesario tener la misma resolución para trabajar con los lotes de datos (batch) de las redes neuronales artificiales. Tampoco se ha usado la resolución original de las imágenes que iban de 520x317 hasta 1280x960 debido a que imágenes de alta calidad pueden requerir muchos recursos y tiempo de procesamiento para

entrenar los modelos de redes neuronales dado que realizan varios barridos de los datos para obtener los mejores resultados.

Usando Keras, se creó un modelo secuencial que permite ir añadiendo capas al modelo. Se configuró el modelo y se procedió a entrenarlo con el conjunto de datos de entrenamiento. Para evitar el sobreajuste del modelo se aplicó la técnica Dropout propuesta por (Srivastava et al. 2014) en donde, de forma aleatoria, se descartan ciertas unidades (neuronas artificiales) y sus conexiones durante el entrenamiento previniendo así que las unidades se adapten demasiado a los datos. Una vez que se encuentra entrenado el modelo se procede a evaluar el mismo con el conjunto de datos de pruebas y se obtiene el nivel de certeza del mismo. Posteriormente, se configuró un script para que acceda a nuevas imágenes e identifique si eran vehículos y los clasifique según su tipo: Bus, Taxi, u Otro.

Finalmente, se guardó el modelo en un archivo h5 que después fue convertido en un búfer plano comprimido usando TensorFlow Lite para ser utilizado en una aplicación móvil que permita probar el modelo en tiempo real utilizando la cámara de video de un smartphone. La aplicación activa la cámara de video y dependiendo de lo que se enfoca realizará la detección de vehículos y su correspondiente clasificación.

### Resultados.

En la primera etapa de la investigación se recolectaron 146 imágenes de distintos vehículos que transitaban por las vías de la ciudad de Riobamba. Estas imágenes se clasificaron de forma manual según el tipo de vehículo (Bus, Taxi, Otro) y posteriormente se agruparon en un conjunto de datos de entrenamiento y otro de pruebas como se indica en la Tabla 1: Conjuntos de datos utilizados en la generación del modelo Tabla 1. Usando la función ImageDataGenerator de la librería keras\_preprocessing.image se importó las 146 imágenes en Python redimensionándolas a 256x256 píxeles en RGB.

**Tabla 1:** Conjuntos de datos utilizados en la generación del modelo

Conjunto de datos	Categoría	Número de imágenes
Entrenamiento	Bus	42
	Taxi	30
	Otro	39
Subtotal		111
Pruebas	Bus	17
	Taxi	10
	Otro	8

Subtotal	35
Total de imágenes	146

**Fuente:** Elaboración propia.

Se creó un modelo de redes neuronales usando la librería keras. La configuración del modelo se muestra en la Figura 1. Como se puede observar en la configuración, se usa transferencia de conocimiento debido a que se utiliza inicialmente el modelo vgg16 propuesto por (Simonyan y Zisserman 2015) el cual ha demostrado que se generaliza bien a una amplia gama de tareas relacionadas con la clasificación de imágenes y a otros conjuntos de datos diferentes a los que usaron cuando se creó este modelo. En un principio se trató de re-entrenar las capas pero por la falta de recursos computacionales se desistió de esta opción y se mantuvo los pesos de las unidades pre-entrenadas del modelo vgg16. De este modelo se remueve la última capa y se configura a cada una de las 21 capas conectadas densamente y convolucionales como no entrenables, para evitar la actualización de los pesos de las unidades debido a que el proceso de entrenamiento para este modelo puede tomar mucho tiempo y requiere de muchos recursos. Después se utiliza una capa Flatten para cambiar la forma del tensor de salida del modelo vgg16 a una forma que es igual al número de elementos contenidos en el tensor. Un tensor es un arreglo multidimensional el cual es usado para representar los datos del modelo. Luego se agrega una capa conectada densamente con 128 unidades y con la función de activación Unidad Lineal Rectificada (RELU). Después se agrega una capa Dropout para evitar el sobreajuste (overfitting) del modelo a los datos del conjunto de entrenamiento con una proporción de 0.4, que indica que el 40% de las unidades serán descartadas en cada paso durante el tiempo del entrenamiento. Finalmente, se agrega una capa densamente conectada con 3 unidades (una unidad por cada clase a predecir) y la función de activación SoftMax para realizar la predicción en base a la selección de la clase que tenga mayor probabilidad.

**Figura 1:** Resumen del modelo de redes neuronales

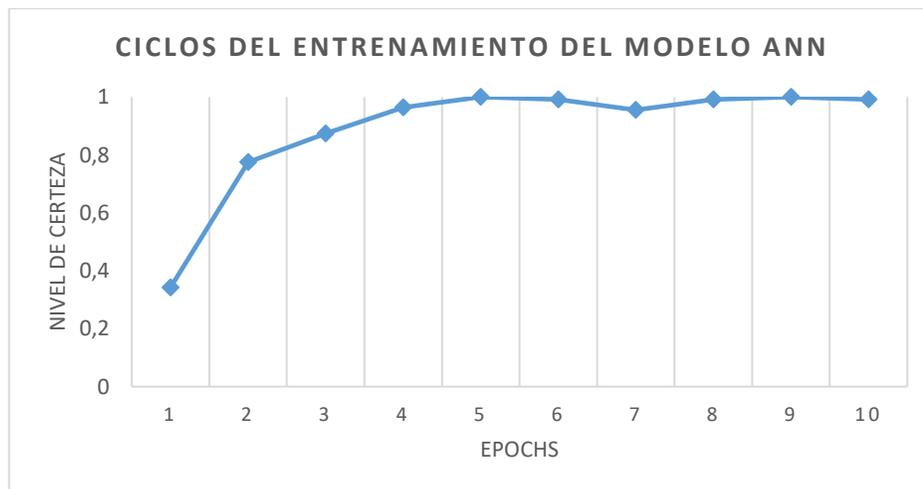
```

Model: "sequential_1"
-----
Layer (type)                Output Shape              Param #
-----
vgg16 (Model)                (None, 8, 8, 512)        14714688
-----
flatten_1 (Flatten)          (None, 32768)             0
-----
dense_2 (Dense)              (None, 128)               4194432
-----
dropout_1 (Dropout)          (None, 128)               0
-----
dense_3 (Dense)              (None, 3)                 387
-----
Total params: 18,909,507
Trainable params: 4,194,819
    
```

**Fuente:** Elaboración propia.

Cuando se concluyó la configuración de las capas, se compiló el modelo usando el optimizador que implementa el algoritmo de Adam propuesto por (Kingma y Ba 2015), y para la métrica de evaluación se escogió el nivel de certeza. Con el modelo compilado, se procedió a entrenarlo en 10 ciclos (epochs). Durante el entrenamiento, el modelo empezó con un nivel de certeza del 0.3191 y se mantuvo mejorando hasta el quinto epoch donde alcanzó un nivel de certeza del 0.9640, después se mantuvo mejorando levemente para finalmente obtener un nivel de certeza de 0.9910 en el conjunto de datos de entrenamiento como se puede observar en la Figura 2. El proceso de entrenamiento tardó 11 minutos en un computador con procesador core i7 de 2.5 GHz y memoria RAM de 12 GB. Python utilizó a lo sumo 33% del procesador y 3.5 GB de memoria RAM durante el entrenamiento. Una vez que se obtuvo el modelo entrenado, se procedió a evaluar el modelo con el conjunto de datos de pruebas, en este caso se alcanzó un nivel de certeza de 0.94285715 que vendría a indicar que aproximadamente el 94.29% de las veces el modelo clasifica de forma correcta un vehículo como Bus, Taxi u Otro. Como se puede observar el nivel de certeza del modelo es bastante alto lo que nos puede llevar a concluir que es factible el uso de redes neuronales en la detección de vehículos y posterior conteo de los mismos. El modelo generado fue exportado desde Python como un archivo con extensión h5.

**Figura 2:** Proceso de entrenamiento del modelo de redes neuronales artificiales con 10 epochs



**Fuente:** Elaboración propia.

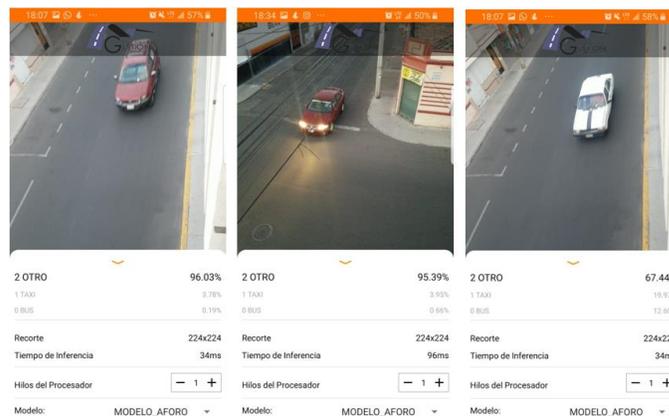
Lo siguiente que se desea determinar, es si el modelo de redes neuronales necesita algún tipo de calibración y/o configuración de los dispositivos para obtener una buena predicción. Por lo tanto, se probó el modelo en una aplicación móvil en tiempo real para determinar si el modelo se ve afectado por el ángulo de la cámara. Por lo que, fue necesario cargar el modelo en un dispositivo móvil, para esto se transformó el modelo a TensorFlow Lite usando las herramientas de conversión que proporciona TensorFlow. Esta transformación convierte el modelo en un archivo comprimido FlatBuffer, por lo tanto, fue necesario evaluar nuevamente el modelo con extensión

tflite. En la evaluación del modelo comprimido se obtuvo el mismo nivel de certeza del modelo sin comprimir 94.29%. Entonces se procedió a generar la aplicación móvil en base a las aplicaciones de código abierto que presenta Tensor Flow Lite. Durante las pruebas se pudo determinar que el modelo puede identificar los vehículos desde diferentes ángulos sin necesidad de calibración como se muestra en la **¡Error! No se encuentra el origen de la referencia.** También es importante resaltar que, durante las pruebas, el tiempo que le tomó a un smartphone realizar la clasificación (tiempo de inferencia) fue de 96 milisegundos, que es bastante rápido para los recursos que posee comparado con un computador.

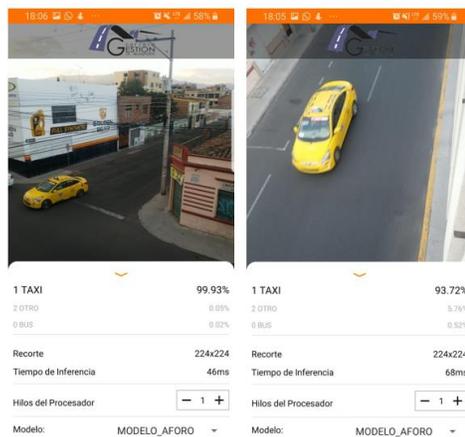
**Figura 3:** a) Detección de buses usando modelo de redes neuronales cargado en aplicación móvil. b) Detección de taxis usando modelo de redes neuronales cargado en aplicación móvil. c) Detección de otros vehículos usando modelo de redes neuronales cargado en aplicación móvil



a)



b)



c)

**Fuente:** Elaboración propia.

El archivo con extensión h5 que contiene el modelo de redes neuronales generado para el computador tiene un tamaño de 106.706 KB y el archivo con extensión tflite que contiene el modelo de redes neuronales para dispositivos móviles tiene un tamaño de 18.488 KB. Como se puede notar, el tamaño de los archivos es bastante pequeño que puede ser compartido fácilmente y puede ser utilizado en una aplicación móvil sin necesidad de consumir muchos recursos.

### Discusión.

Durante el desarrollo de este estudio se ha obtenido un modelo de redes neuronales que permite clasificar vehículos con el fin de demostrar la factibilidad de la utilización de redes neuronales artificiales en la automatización del aforo vehicular y clasificación de vehículos de transporte público por medio de video captado por cámaras sin la necesidad de una configuración o calibración inicial. Trabajos futuros se puede usar el modelo propuesto en este estudio como base para crear un modelo más sofisticado que incluya métodos como el Mask R-CNN (He et al. 2018) que permite enmarcar cada objeto que se desea detectar. Por otra parte, (Avery, Wang, y Rutherford 2004) proponen un algoritmo para la detección de vehículos, sin embargo, este algoritmo requiere de cierta configuración inicial para crear una línea digital sobre la cual deben pasar los vehículos para ser detectados siempre y cuando la variación de los colores en la línea sea mayor a un umbral configurado inicialmente; esta configuración inicial muchas de las veces puede ser un proceso tedioso que debe pasar por pruebas con el fin de seleccionar la mejor configuración; en este caso la existencia de un sistema basado en redes neuronales que no requiera configuración inicial puede ser de gran utilidad para los usuarios finales porque puede minimizar costos y tiempos de instalación. (Chintalacheruvu y Muthukumar 2012) proponen un sistema para la detección de vehículos basado en el algoritmo de detección de esquinas de Harris-Stephen que se encuentra implementando en la librería OpenCV, esta solución permite identificar a los vehículos mediante la detección de puntos de interés en los mismos separando las vías en carriles, por lo cual, se necesita de una calibración inicial del sistema en base al ángulo de la cámara para

determinar cada uno de los carriles y la dirección de los vehículos; este proceso puede llegar a ser abrumador dado que se lo tiene que realizar por cada cámara de video dado que una variación en el ángulo de la cámara provocaría una des calibración del sistema. Por lo tanto, la propuesta de utilizar modelos de redes neuronales que pueden trabajar sin previa calibración sigue siendo una alternativa que podría mejorar la automatización del aforo vehicular simplemente accediendo al video de los CCTV que ya se encuentran en muchas ciudades del mundo funcionando desde hace varios años.

### Conclusiones:

- En este trabajo se analiza la factibilidad de utilizar modelos de predicción de redes neuronales artificiales en la automatización del aforo y clasificación vehicular, por lo cual, se ha desarrollado satisfactoriamente un modelo de ANN que permite identificar vehículos y clasificarlos en una de tres categorías con un nivel de certeza del 94.29%. El tiempo de entrenamiento del modelo fue de 11 minutos que es relativamente rápido debido a que se usó como base el modelo pre-entrenado vgg16 que contiene 21 capas que incluyen capas densamente conectadas y convolucionales. También es importante notar que los modelos generados ocupan muy poco espacio y eso les permite trabajar tanto en computadores como en dispositivos móviles teniendo un tiempo de inferencia máximo de 96 milisegundos en dispositivos móviles.
- La aplicación móvil permitió probar la detección y clasificación de vehículos en tiempo real y con variaciones del ángulo de la cámara para demostrar que los modelos de redes neuronales pueden realizar inferencias sin necesidad de una calibración previa en el sistema o las cámaras de video siempre y cuando se entrene al modelo con una basta cantidad de datos. En este estudio la mayoría de los datos utilizados correspondían a fotos de vehículos tomadas desde diferentes perspectivas y alturas.
- Por último, se concluye que, debido a que se ha demostrado la factibilidad del uso de redes neuronales artificiales en la automatización del aforo y clasificación vehicular, se debe profundizar en el perfeccionamiento de modelos de ANN para incluir métodos más complejos como Mask R-CNN que permitan incrementar el nivel de certeza y funcionen cuando existen varios vehículos al mismo tiempo.

### Referencias bibliográficas.

- Alpizar, Carlos, y Arnaldo del Risco Sánchez. 2012. "Diseño de un Sensor de Flujo Vehicular basado en Lazo Inductivo". *Revista Científica de Ingeniería Electrónica, Automática y Comunicaciones* 33(1): 33–44.
- Avery, Ryan P., Yinhai Wang, y G. Scott Rutherford. 2004. "Length-based vehicle classification using images from uncalibrated video cameras". *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC: 737–42.*
- BETANCOURT, JUAN ESTEBAN. 2019. "Diseño de una plataforma de conteo automatizado para el aforo vehicular con base en la clasificación de (ejes y peso), para vías secundarias, como insumo de diseño de pavimentos flexibles". UNIVERSIDAD PILOTO DE

COLOMBIA 1(Novembre). <http://dx.doi.org/10.1101/843326>.

- Chintalacheruvu, Naveen, y Venkatesan Muthukumar. 2012. "Video Based Vehicle Detection and its Application in Intelligent Transportation Systems". *Journal of Transportation Technologies* 02(04): 305–14.
- Harris, C., y M. Stephens. 2013. "A Combined Corner and Edge Detector". *Alvey vision conference*: 23.1-23.6.
- He, Kaiming, Georgia Gkioxari, Piotr Dollár, y Ross Girshick. 2018. "Mask R-CNN". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42(2): 386–97.
- Kingma, Diederik P, y Jimmy Lei Ba. 2015. "Adam: A method for stochastic optimization". En *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, , 1–15.
- Minge, Erik, Scott Petersen, y Jerry Kotzenmacher. 2011. "Evaluation of nonintrusive technologies for traffic detection, phase 3". *Transportation Research Record* (2256): 95–103.
- Mishra, Manish, y Monika Srivastava. 2014. "A view of artificial neural network". En *2014 International Conference on Advances in Engineering & Technology Research (ICAETR-2014)*, , 1–3.
- MONETTI, Julio, Micaela CONTRERAS, Martín NAVARRO, y Gerardo SEVILLANO. 2018. "PROPUESTA DE RECOLECCIÓN DE DATOS PARA AFORO VEHICULAR". *XX Workshop de Investigadores en Ciencias de la Computación - WICC 2018 : libro de actas*: 6,13. <http://hdl.handle.net/10915/67063>.
- Sharma, Satish C. 1983. "Minimizing Cost of Manual Traffic Counts: Canadian Example". *Transportation Research Record* 905(2): 1–7.
- Simonyan, Karen, y Andrew Zisserman. 2015. "Very deep convolutional networks for large-scale image recognition". *ICLR*.
- Slinn, Mike. 2006. "Traffic Engineering Design". *Traffic Engineering Design*.
- Srivastava, Nitish et al. 2014. "Dropout: a simple way to prevent neural networks from overfitting". *The journal of machine learning research* 15(1): 1929–58.
- Zheng, Pengjun, y McDonad Mike. 2012. "An Investigation on the Manual Traffic Count Accuracy". *Procedia - Social and Behavioral Sciences* 43: 226–31. <http://dx.doi.org/10.1016/j.sbspro.2012.04.095>.

**PARA CITAR EL ARTÍCULO INDEXADO.**

Moreno Vallejo, P. X., Bastidas Guacho, G. K., & Moreno Costales, P. R. (2020). Estudio de factibilidad del uso de modelos de redes neuronales artificiales en la automatización del aforo y clasificación vehicular del transporte público. *ConcienciaDigital*, 3(3), 528-540. <https://doi.org/10.33262/concienciadigital.v3i3.1355>



El artículo que se publica es de exclusiva responsabilidad de los autores y no necesariamente reflejan el pensamiento de la **Revista Conciencia Digital**.

El artículo queda en propiedad de la revista y, por tanto, su publicación parcial y/o total en otro medio tiene que ser autorizado por el director de la **Revista Conciencia Digital**.

