

**DISEÑO DE UN ALGORITMO DE RECONOCIMIENTO DE
PLACAS VEHICULARES ECUATORIANAS USANDO
REDES NEURONALES CONVOLUCIONALES**
*DESIGN OF AN ECUADORIAN VEHICLE LICENSE PLATE
RECOGNITION ALGORITHM USING CONVOLUTIONAL NEURAL
NETWORKS*

<https://doi.org/10.5281/zenodo.4147629>

AUTORES: Eduardo Barbecho Bautista^{1*}

Martín Zhindón Mora²

DIRECCIÓN PARA CORRESPONDENCIA: eduardo.barbecho@psg.ucacue.edu.ec

Fecha de recepción: 20 / 02 / 2020

Fecha de aceptación: 11 / 09 / 2020

RESUMEN

La detección de vehículos robados en Ecuador es una de las tareas más importantes para las agencias de seguridad del país. Una de las formas de aumentar la tasa de detección y recuperación de vehículos es a través de aplicaciones inteligentes, basadas en inteligencia artificial y redes neuronales. Estas aplicaciones permiten el desarrollo de nuevas técnicas de reconocimiento de placas vehiculares (LPR). En este artículo, proponemos un modelo para detectar placas de automóviles en movimiento utilizando redes neuronales convolucionales, primero se realiza un etiquetado manual sobre las imágenes de los vehículos, identificando la ubicación de la placa y caracteres dentro de la imagen. Esta información se introduce en una arquitectura de redes neuronales convolucionales para entrenamiento y pruebas. La arquitectura de la red neuronal es desarrollada por Google, COCO Inception V2, y se utiliza como base de entrenamiento, de donde se obtiene un modelo propio entrenado para placas

^{1*}Estudiante de Maestría en Tecnologías de la Información, Universidad Católica de Cuenca, eduardo.barbecho@psg.ucacue.edu.ec

²Magister en Geomática, Universidad Católica de Cuenca, mgzhindonm@ucacue.edu.ec

ecuatorianas. Los resultados experimentales muestran que el presente trabajo alcanza una precisión de reconocimiento favorable de 85.1 % en términos del conjunto de datos de fotos de entrenamiento de placas ecuatorianas. Vale la pena mencionar el uso exclusivo de software de código abierto para el desarrollo.

Palabras clave: Convolución, neurona, placas vehiculares, inteligencia artificial, aprendizaje.

ABSTRACT

Detecting stolen vehicles in Ecuador is one of the important tasks for the country's security agencies. One of the ways to increase the detection and recovery rate of vehicles is through intelligent applications, based on artificial intelligence and neural networks. These applications allowed the development of new vehicle license plate recognition (LPR) techniques. In this article, we propose a model to detect moving car plates using convolutional neural networks. First, a tagged manual is made on the images of the vehicle, identifying the location of the plate and the characters within the image. This information is presented in a convolutional neural network architecture for training and testing. The network architecture developed by Google, COCO Inception V2, is used as a training base, where we get our own trained model for Ecuadorian vehicles plates. The experimental results find that the present work achieves a favorable recognition precision of 85.1 in terms of the Ecuadorian plate training photo data set. It is worth specifying the exclusive use of open source software for the development of this work.

Keywords: Convolution, neuron, license plates, artificial intelligence, learning

INTRODUCCIÓN

En Ecuador, según el Ministerio del Interior (2017) se han registrado 13.092 robos. De estos; 9,377 ilícitos corresponden a robos de bienes, accesorios, auto partes de vehículos; mientras que 4,147 altercados, han sido robos de carros. Para el año 2018, según datos de la Policía Judicial del Ecuador, los delitos contra vehículos se han incrementado de manera alarmante, lo que se refleja en las estadísticas de la seguridad ciudadana del país. Más de 4,195 vehículos han sido robados en todo el país hasta el mes de octubre 2018. El alto nivel de denuncias, permite que las circunstancias en que se producen estos delitos sean ampliamente conocidas por las autoridades, lo que, en principio, debería facilitar el diseño e

implementación de políticas públicas específicas para contrarrestar su alta ocurrencia. Asimismo, se debe tener en cuenta que este incremento de los hechos delictivos y de los grados de violencia está relacionado con el surgimiento y consolidación de mercados ilegales de transacción económica de piezas y vehículos robados.

El objetivo del presente trabajo es desarrollar un modelo de algoritmos de visión artificial que sea usado como herramienta para mejorar la tasa de recuperación vehicular, a través de la identificación de placas vehiculares ecuatorianas, empleando la arquitectura de redes neuronales que mejor se adapte al presente. Existen varios modelos desarrollados en el medio, por lo que la presente investigación busca identificar la mejor opción que se aplique a detección de placas ecuatorianas. En el cambio de reconocimiento de objetos y caracteres las redes neuronales convolucionales son las más empleadas. Existen varias arquitecturas para redes neuronales convolucionales profundas (CNN) desarrolladas para para ubicar, segmentar e identificar objetos como placas vehiculares o caracteres. Uno de los métodos más conocidos es el método de detección de bordes (Abolghasemi *et al.* 2009) donde es necesario las etapas de filtrado, extracción de las características, segmentación y reconocimiento de caracteres. También Y. P. Huang, C. H. Chen, Y. T. Chang (2009), emplean el método de ventana deslizante, que agrega dos etapas de segmentación horizontal y vertical con desplazamiento de la ventana para corregir el ángulo en el cual es adquirida la imagen, esto debido a que se aplica en motos y las placas de estas se encuentran más cercanas al piso. Guo - Liu (2008) utilizaron la ecuación del histograma para mejorar el contraste entre los caracteres y el fondo, las propiedades de textura para ubicar la placa y la transformación de Hough para corregir la inclinación. Los métodos de Jaskirat S. Barat B. (2019) detectaron la ubicación de la matrícula del vehículo iraní mediante el uso del color de destino y la coincidencia de plantillas modificadas, además los caracteres fueron reconocidos por el árbol de decisión SVM con núcleo polinomial homogéneo de quinto grado. Todos los trabajos anteriores necesitan un desarrollo matemático complejo para describir el funcionamiento de las diferentes redes neuronales desarrolladas. Sin embargo, en el año 2016 en el concurso realizado por MS COCO (conjunto de datos de detección, segmentación y sub titulación de objetos a gran escala), Google a través de su grupo G-RMI, (Google Research and Machine Intelligence), ganó el

primer lugar en el desafío de detección de 2016. Desarrolló seis tipos de arquitecturas de CNN. Dentro de estas destaca Inception V2 por su performance. El desarrollo de la red Inception es un parámetro fundamental en el desarrollo de clasificadores de CNN, por tanto de este trabajo. Existen varias versiones de la red como la Inception v1, Inception v2, Inception v3, Inception v4 e Inception-ResNet. Cada versión es una mejora de la anterior. Cabe mencionar que, dependiendo de los datos, una versión inferior puede funcionar mejor que una superior. Uno de los mayores problemas al realizar el análisis y clasificación de objetos es que el objeto dentro de la imagen puede cambiar de tamaño y posición. Debido a esta gran variación en la ubicación de la información, elegir el tamaño correcto del núcleo para la operación de convolución es difícil. Además, una elevada cantidad de convoluciones es de alto costo computacional. La versión v1 de Inception diseña la red con filtros de varios tamaños que funcionen al mismo nivel, haciendo la red más ancha en lugar de más profunda. Además, también se realiza la concatenación de las salidas para enviar al siguiente módulo de inicio. Para hacer las redes más livianas, se limita el número de canales de entrada al agregar una convolución 1x1 adicional. La figura [1] muestra cómo se realiza la convolución en una entrada.

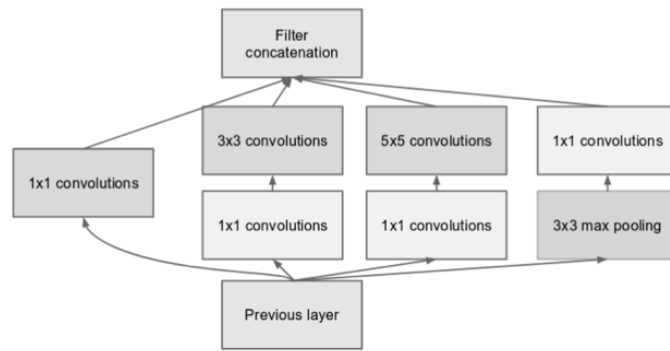


Figura 1. Módulo Inception V.1, con dimensiones reducidas.

Fuente: Szegedy, Vanhoucke - Rethinking the Inception Architecture for Computer Vision

Elaboración: El autor

Usando este modelo reducido, se construye la arquitectura de red neuronal conocida como GoogLeNet. El problema de este clasificador es que al ser muy profundo presenta problemas de pérdida de información. Para solventar este problema, la segunda versión o Inception v2 implementa técnicas para reducir el cuello de botella. Este concepto significa que no se altera significativamente las dimensiones de la entrada a la red. Se emplean tres

métodos de factorización inteligente, dando como resultado operaciones más eficientes en términos de complejidad computacional. Los operaciones de factorización reducen hasta en un 30 % la complejidad computacional del modelo según lo indica Szegedy - Vanhoucke (2015). La figura [2] muestra los cambios mencionados.

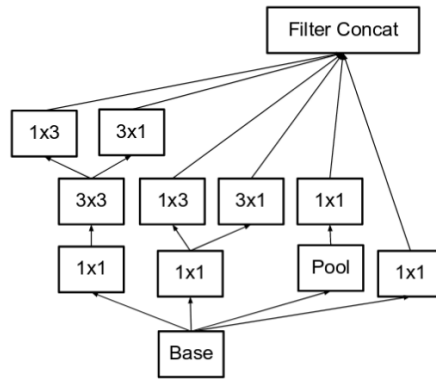


Figura 2. Modelo Inception V.2

Fuente: Szegedy, Vanhoucke - Rethinking the Inception Architecture for Computer Vision

Elaboración: El autor

Los tres principios mencionados se usan para construir tres tipos diferentes de módulos iniciales Inception A, B y C. La arquitectura se muestra en la figura [3].

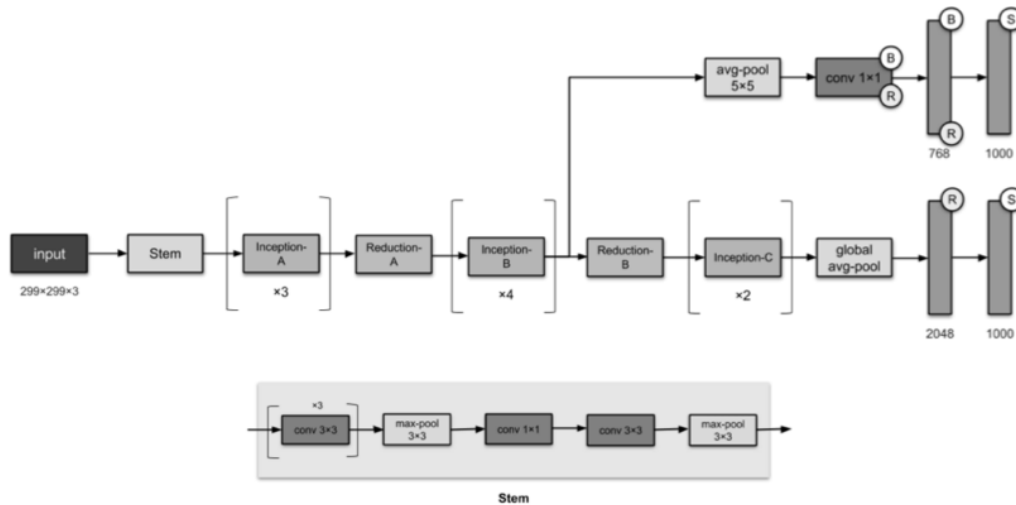


Figura 3. Arquitectura completa Inception v2 / v3

Fuente: [Raimi Karim](#) - A compiled visualisation of the common convolutional neural networks

Elaboración: El autor

Dentro del campo de la investigación de la detección de placas vehiculares ecuatorianas, son escasos los trabajos enfocados en algoritmos de redes neuronales convolucionales

profundas. La presente investigación, busca determinar que el modelo generado por Google, COCO Inception v2, brinda un porcentaje aceptable de confiabilidad en la detección de placas vehiculares ecuatorianas dentro de una imagen, y de la misma manera el modelo actúa de manera adecuada para la identificación de caracteres de las placas. La figura [4] muestra que el modelo seleccionado como base, tiene un rendimiento bastante aceptable, por lo que se emplea esta arquitectura como base de entrenamiento para la obtención del nuevo modelo de red neuronal para placa ecuatoriana.

Model	Top-1 accuracy
VGG-16	71.0
MobileNet	71.1
Inception V2	73.9
ResNet-101	76.4
Inception V3	78.0
Inception Resnet V2	80.4

Figura 4. Modelo COCO Inception V2 - ganador del concurso de detección de objetos.

Fuente: [Sik-Ho Tsang](#) - A Guide to Select a Detection Architecture: Faster R-CNN, R-FCN and SSD.

Elaboración: El autor

El presente documento está distribuido en cuatro secciones. En la primera sección se discute las generalidades trabajos previos e introducción. En la segunda sección se detalla la metodología, y las etapas para el desarrollo del modelo. Los resultados de la investigación se muestran en la tercera sección. Finalmente, en la cuarta sección se detalla, las conclusiones, recomendaciones y trabajos futuros.

METODOLOGÍA

Para el desarrollo del presente trabajo se ha determinado once fases a realizar, las cuales consisten en depuración de las imágenes, preparación de plantillas de entrenamiento, el entrenamiento en sí, y la validación. La figura [5] muestra en detalle todo el procedimiento.

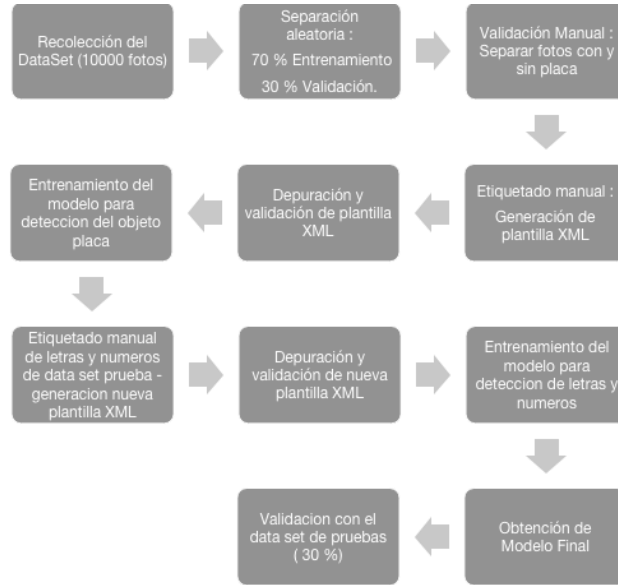


Figura 5. Etapas del reconocimiento de placas vehiculares ecuatorianas.

Fuente: Autor

El proceso empieza con un dataset de 10 000 fotos desde el servidor lector de placas del Sistema Integrado de Seguridad ECU911 Ecuador, para entrenamiento y validación. Cabe mencionar que las circunstancias bajo las cuales se toman las fotos son el vehículo en movimiento, con una velocidad promedio de 80 Km/h, a una altura de 6 metros y distancia horizontal de 30 metros. Como siguiente paso, es necesario separar el set de fotos en dos grupos; entrenamiento y pruebas, en porcentaje de 70% - 30% respectivamente. La separación se realiza de manera aleatoria, para que el sistema aprenda bajo las diferentes situaciones externas, como luz de día y noche, el sol de la costa o el clima nubloso de la sierra. Las fotos del dataset son de cámaras ubicadas alrededor de todo el territorio ecuatoriano. El siguiente paso es etiquetar de manera manual las 7000 fotos de entrenamiento, ubicando las coordenadas de la zona que contiene el objeto placa. Para este procedimiento se emplea la herramienta de código libre Imglab, la cual se muestra en la figura [6]. Este proceso genera una plantilla XML que contiene los datos de coordenadas.



Figura 6. Herramienta ImgLabel para etiquetado manual del dataset de fotos.

Fuente: Autor

Luego, se depura la plantilla y se le da formato para prepararla para el entrenamiento. En la figura [7] se muestra la plantilla generada para una imagen en donde se identifica las coordenadas del objeto placa. La siguiente etapa es el entrenamiento y obtención del nuevo modelo que permita detectar el objeto placa, resultado del aprendizaje del dataset de fotos de placas ecuatorianas. En esta etapa se emplea la arquitectura de red de Google. Luego de generar el modelo de detección del objeto placa, se procede a identificar los caracteres de la foto. Para esto, se realiza el mismo proceso anterior de etiquetado manual, esta vez enfocando en los caracteres de la imagen.



Figura 7. Plantilla XML generada por el etiquetado manual, previo al entrenamiento.

Fuente: Autor

Se genera una nueva plantilla XML con las coordenadas de cada una de las letras dentro de la imagen. Se realiza el recorte o crop de cada una de las letras y números tomando en cuenta las coordenadas obtenidas en la plantilla XML. Se genera un nuevo data set con letras y números, el cual será el insumo de entrenamiento para el segundo modelo. Se procede a entrenar el modelo con la misma arquitectura InceptionV2. Como resultado se obtiene un modelo que identifica los caracteres de las placas ecuatorianas. Se procede a enlazar los dos modelos pre entrenados dentro de la arquitectura de Google para obtener un modelo final. Finalmente, se valida la precisión del modelo realizando las pruebas sobre el 30 % del data set de validación.

RESULTADOS

Para evaluar el rendimiento del modelo propuesto, se experimenta con el data set de validación, es decir con el 30 % de las 10000 fotos. El Dataset contiene imágenes que fueron adquiridas en condiciones de día y noche, con los vehículos en movimiento. El primer se entrena un modelo inicial que obtiene como resultado la ubicación del objeto placa. Las fotos del dataset muestran matrículas con datos en diferentes condiciones, como desorden de fondo, rotación y cambio de escala. Dichos factores alteran la precisión del modelo. Esta información sirve de sustento para el segundo algoritmo. Se realiza el etiquetado manual y recorte de los caracteres de la placa, dicha información se anexa a la información del primer modelo y se prepara la plantilla para el entrenamiento del segundo modelo. Se procede a entrenar un segundo modelo que aprende a identificar que caracteres aparecen en las imágenes de las placas. La precisión final del modelo y los algoritmos que lo conforman es del 85 %, en base al dataset de 10000 fotos. La fase de la detección de la placa obtuvo un porcentaje del 84 %, mientras que en la detección de caracteres se obtuvo una precisión del 86 %. Cabe mencionar que las placas con más fallas de detección son aquellas que son de color tomate, es decir de transporte público ecuatoriano. Además, el modelo no responde de la mejor manera cuando se localiza dentro de la imagen más de una placa. En la figura [8] se muestra el proceso de segmentación del objeto placa, luego los caracteres y el reconocimiento de los caracteres aprendidos.



Figura 8. Detección de placa y de caracteres.

Fuente: Autor

DISCUSIÓN

Para el presente trabajo las fotos son tomadas por cámaras del SISECU911 que se encuentran instaladas en carreteras con vehículos en movimiento y de alta velocidad. Además, las cámaras se encuentran a una distancia considerable de la placa del vehículo, lo cual influye en la precisión del modelo. En la figura [9] se resume los resultados de la evaluación del presente modelo en comparación con trabajos y técnicas tomadas como referencia en el presente trabajo.

Modelo	Precision (%)	Distancia de vehículo
Mostafa-Shahbahrami	94%	3 m
Yo-Ping , Chien-Hung	93.90%	3 m
Singh - Bharat	95%	3 m
Presente modelo	85%	30 m

Figura 9. Comparativa y evaluación del modelo.

Fuente: Autor

El modelo desarrollado en el presente documento muestra una precisión menor que los modelos desarrollados en trabajos anteriores. Esto se debe, a que en este caso, el data set de fotos que sirve como sustento para el entrenamiento del modelo presentan algunas diferencias. Las fotos del SISECU911 que entrenaron al presente modelo, son tomadas a una mayor distancia del suelo y del vehículo. Adicionalmente, las imágenes son

capturadas con los vehículos en movimiento a grandes velocidades por lo que el ángulo, nitidez y ubicación de las placas dentro de la foto son más complejos de procesar.

CONCLUSIONES

Se desarrolla un modelo de detección de placas vehiculares ecuatorianas que sirve para mejorar la tasa de recuperación vehicular. Se concluye que la arquitectura desarrollada por Google, Inception V.2, es una buena opción para identificar objetos de placas vehiculares ecuatorianas. La precisión del modelo que se obtiene depende de la calidad y el tamaño de la información que alimente al algoritmo para su entrenamiento. En este trabajo se emplearon 10000 fotos en diferentes condiciones de día, luz y clima. La precisión del algoritmo mejora si se aumenta el número de fotos del dataset de entrenamiento.

REFERENCIAS BIBLIOGRÁFICAS

- Guo, Jing Ming, y Yun Fu Liu. 2008. "License plate localization and character segmentation with feedback self-learning and hybrid binarization techniques". *IEEE Transactions on Vehicular Technology* 57 (3): 1417–24. <https://doi.org/10.1109/TVT.2007.909284>.
- Huang, Yo Ping, Chien Hung Chen, Yueh Tsun Chang, y Frode Eika Sandnes. 2009. "An intelligent strategy for checking the annual inspection status of motorcycles based on license plate recognition". *Expert Systems with Applications* 36 (5): 9260–67. <https://doi.org/10.1016/j.eswa.2008.12.006>.
- Mobarhan, Mostafa Ayoubi, Asadollah Shahbahrami, Saman Parva, Mina Naghash Asadi, y Atefeh Ahmadnya Khajekini. s/f. "AISC 166 - A License Plate Detection Algorithm Using Edge Features". *Eng. & Appl.* Vol. 166.
- Singh, Jaskirat, y Bharat Bhushan. 2020. "Real Time Indian License Plate Detection using Deep Neural Networks and Optical Character Recognition using LSTM Tesseract", 347–52. <https://doi.org/10.1109/icccis48478.2019.8974469>.
- Yuan, Yule, Wenbin Zou, Yong Zhao, Xinan Wang, Xuefeng Hu, y Nikos Komodakis. 2017. "A Robust and Efficient Approach to License Plate Detection". *IEEE Transactions on Image Processing* 26 (3): 1102–14. <https://doi.org/10.1109/TIP.2016.2631901>.
- Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, y Zbigniew Wojna. 2016. "Rethinking the Inception Architecture for Computer Vision". Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2016-Decem: 2818–26. <https://doi.org/10.1109/CVPR.2016.308>.