

Segmentación de placas vehiculares usando Haar-AdaBoost y Clustering

José Hernández Santiago^{1,2}, José Sergio Ruiz Castilla², Carlos Hiram Moreno Montiel¹,
Beatriz Hernández Santiago²

¹ Tecnológico de Estudios Superiores de Chimalhuacán,
México

² Universidad Autónoma del Estado de México, Posgrado e Investigación,
Texcoco, Estado de México,
México

{josehernandez, carlosmoreno}@teschi.edu.mx,
jhernandezs@uaemex.mx,jsergioruizc@gmail.com,betty_hsb@hotmai.com

Resumen. Actualmente los sistemas de seguridad vial cuentan con cámaras IP para monitorear las carreteras, sin embargo el análisis de estos representa un problema muy importante ya que es el personal a cargo quienes realizan esta labor. Un primer paso para desarrollar un sistema de visión que implemente técnicas de inteligencia artificial para llevar el registro de los automóviles que circulan es poder extraer la imagen de la placa del auto a partir del video de las cámaras de seguridad, después segmentarla para extraer las imágenes de los caracteres para finalmente poder realizar la clasificación y obtener el número de serie del automóvil. En este artículo, se presenta un nuevo enfoque para generalizar el problema de la segmentación de las placas de automóviles usando primero AdaBoost para ubicar la placa dentro de la imagen, después el clasificador SVM para descartar falsos positivos y Clustering para ubicar los caracteres dentro de la placa. El método propuesto permite segmentar las placas a pesar de los diversos diseños que existen en México, su ubicación en el auto y la vista frontal o lateral de la cámara.

Palabras clave: Características de Haar, AdaBoost, clustering, máquinas de vectores soporte, segmentación de placas vehiculares.

Licence Plate Segmentation by Haar-AdaBoost and Clustering

Abstract. In the Mexican Republic, vial safety systems currently have IP cameras to monitor roads. However, the analysis of monitoring system represents a several problem because it works in a personal human. Through three steps can develop a vision system to generate a record video of the cars that circulate. With artificial intelligence techniques the image of the plate and the car could be extracted from the video of the surveillance cameras, the first step. As a second step, the image must be

segmented to extract the most important characteristics and generate a comparative model of the plates. Finally, as a third step to perform identification, a classification is needed to extract the car's serial number. In the present article a novel approach is shown to generalize the problem of segmentation of automobile license plates. Using in the first instance the AdaBoost method to localize the plate inside the image. Then an SVM classifier is used to detect false positives and clustering to locate the characters inside the plate. The generated method the plates to be segmented regardless of the different designs that exist in the Mexican Republic and location of the auto, tilt, the front or side view of the camera.

Keywords: Haar features, AdaBoost, clustering, support vector machines, licence plate segmentation.

1. Introducción

Actualmente en el Estado de México se reportan 12 mil autos robados, mientras que la cifra asciende a 86 mil durante el último año, representando un problema importante de seguridad. Las instituciones de seguridad pública han implementado sistemas de video vigilancia con cámaras IP, Fig. 1.b, sin embargo, el análisis de los videos almacenados se realiza con el personal disponible, Fig. 1 a, siendo insuficiente para atender todos los incidentes reportados y buscar los autos robados.



a) Monitoreo de la cámara IP



b) Cámara IP

Fig. 1. Sistema de Seguridad.

La detección de placas vehiculares a partir de los videos de seguridad es un problema que se ha abordado de diferentes formas. En [1] el método de segmentación de placas se basa en buscar regiones candidatas donde puede haber caracteres usando Maximally Stable Extremal Region (MSER), obteniendo una precisión de 95% para videos viales y con diferentes niveles de iluminación, sin embargo sus imágenes tienen alta definición y la norma en China estandariza sus placas en fondo azul con texto blanco o fondo amarillo con texto negro, reduciendo la complejidad del problema.

En [2] proponen un algoritmo híbrido aplicando segmentación y extracción de contornos, mapeando cada región candidata a un plano euclidiano y determinando su relevancia a través de una función de costo; las imágenes usadas tienen diferentes iluminaciones, sin embargo la tarea se facilita al usar solo fotos frontales y placas con fondo negro con texto blanco.

En [3] utilizan las líneas como características para ubicar las placas chinas, usando el espacio de color de Munsell y distancia NBS, así como un tratamiento a la imagen con clustering antes de aplicar los filtros; después otro filtro morfológico es empleado para seleccionar las placas candidatas de acuerdo a su forma.

Un RFID es usado en [4] para detectar cuando el automóvil esta frente a la cámara Full HD, autorizando su paso si es que el número de la placa tiene acceso. La extracción de la placa se realiza primero usando un umbral adaptativo, dilatación y búsqueda de contornos cerrados, que al aplicar condiciones como regiones con 20 a 25% de pixeles negros con fondo blanco y rectangularidad de 90%, detecta el contorno como placa más probable con 97% de precisión.

El método presentado en [5] utiliza el seguimiento del vehículo en el video para ocupar la secuencia de frames y crear clusters con las placas parecidas que ayudaran en la clasificación del número de la placa con OpenALPR, mejorando la precisión de 7 a 32% comparado con un sistema que no usa tracking.

En [6] se comparan las técnicas de Redes Neuronales Artificiales Backpropagation (BPNN), Redes de Función de Base Radial (RBF) y Ensamble de Redes Neuronales (ENN) para reconocer placas de automóviles de Malasia respecto a su método propuesto, que compara el histograma de los objetos encontrados en la imagen binarizada para ubicar la placa y clasifica los caracteres usando una ENN que promedia la salida de una BPNN con una RBF; sin embargo las fotos provienen de un estacionamiento, no presentan inclinación o diferente vista y las placas están rotuladas en color blanco sobre fondo negro.

Una Red Neuronal Convolutiva es usada en [7] para clasificar placas chinas, reportando 98.95% de precisión para ubicar la placa, 96.58% de precisión en la segmentación de los caracteres y 98.09% en la clasificación. Para localizar la placa usaron un filtro de color, detección de contornos y un análisis morfológico, sin embargo las fotos probadas son frontales y provienen de estacionamientos. De forma similar en [8] se obtiene 98.42% de precisión en la localización de la placa.

En los artículos anteriormente citados se aplican filtros para poder ubicar la placa dentro de las imágenes, mientras que otros parten de imágenes de placas previamente cortadas para centrarse en la clasificación de los caracteres, además de que las placas incluyen poco ruido por desgaste.

Las características de Haar inicialmente fueron usadas para el tracking de los rostros [9, 10] y extraer características como los ojos [11], pero se ha demostrado que pueden emplearse en otras aplicaciones como en [12] donde las características de Haar y AdaBoost son usados para detectar y seguir los vehículos completos en videos de seguridad con vista superior logrando 93% de precisión mientras que en [13] una técnica similar permite reconocer los logotipos de algunas marcas de vehículos.

Actualmente se han implementado técnicas para el seguimiento de objetos usando boosting, como en [14] donde mejoran el algoritmo AdaBoost con un cumulo de partículas para detectar autos completos a partir de características rectangulares y puntos de control; mientras que en [15] de forma similar logran detectar vehículos con 93.2% de precisión usando Gentle AdaBoost y Patrones Binarios Locales (LBP).

En este artículo se presenta un nuevo enfoque para extraer la placa y los caracteres a partir de las imágenes de video. En la metodología se explica cómo el método propuesto aplica AdaBoost para clasificar usando características de Haar y poder ubicar la placa dentro

de la imagen del video; posteriormente se eliminan los falsos positivos usando una Máquina de Vectores Soporte (SVM); después de extraer la imagen de la placa se detectan las líneas para corregir la inclinación y finalmente se aplica un filtro de umbral por rango de color de forma iterativa hasta encontrar el cluster con todos los caracteres.

En la sección de resultados se puede observar que el método de segmentación propuesto mantiene una buena precisión para ubicar la placa y los caracteres, aunque las placas vehiculares tengan ruido por inclinación, escala, vista de la cámara e imágenes de fondo.

2. Preliminares

a. Características de Haar

Las características de Haar indican la diferencia de intensidades de los píxeles en regiones rectangulares locales permitiendo la detección de bordes (Fig. 2), líneas (Fig. 3) y el centro (Fig. 4). Estas características serán usadas como entradas de un clasificador básico que permitirá detectar objetos de forma rápida.



Fig. 2. Características para detectar bordes.

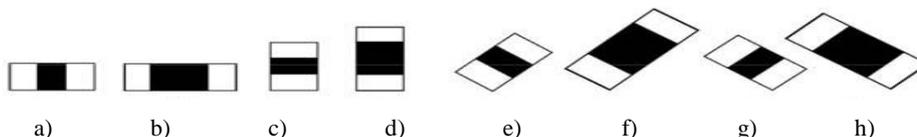


Fig. 3. Características para detectar líneas.

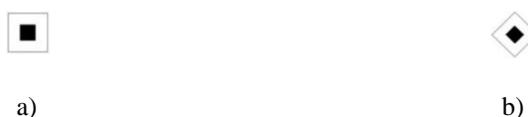


Fig. 4. Características para detectar el centro.

b. AdaBoost

Este método está diseñado para combinar reglas, se basa en la combinación lineal de muchas reglas débiles pero muy precisas, para crear un clasificador muy robusto con un error arbitrariamente bajo en el conjunto de entrenamiento. Estas reglas débiles se aprenden secuencialmente manteniendo una distribución de pesos D_t sobre los ejemplos de entrenamiento. Estos pesos se van actualizando a medida que se adquieren nuevas reglas.

AdaBoost es un algoritmo que pretende obtener una regla de clasificación muy precisa combinando muchos clasificadores débiles, cada uno de los cuales obtiene una precisión

moderada. Este algoritmo trabaja eficientemente con espacios de atributos muy grandes y ha sido aplicado con éxito a muchos problemas prácticos.

Las hipótesis débiles $h_t: X \rightarrow \{-1, +1\}$, se aprenden secuencialmente en cada iteración, sesgándola para clasificar los ejemplos con más dificultad de acuerdo al conjunto de reglas con las que se cuentan en esa iteración, manteniendo un error moderadamente bajo respecto a la distribución de pesos D_t . Inicialmente, la distribución de pesos D_1 es uniforme, y en cada iteración, el algoritmo de boosting incrementa (o decreta) exponencialmente los pesos $D_t(i)$ en función de si $h_t(x_i)$ realiza una buena (o mala) predicción. La combinación final de hipótesis, $h_t: X \rightarrow \{-1, +1\}$, calcula sus predicciones ponderando con pesos los votos de las diferentes hipótesis débiles como se muestra en la ecuación (1):

$$f(x) = \sum_{t=1}^T \alpha_t \cdot h_t(x). \quad (1)$$

Para cada nuevo ejemplo x , el signo de $f(x)$ se interpreta como la clase predicha (-1 o $+1$), y la magnitud $|f(x)|$ como una medida de la confianza de la predicción. En 2001 Viola y Jones inventaron un clasificador basado en las características de Haar que permite detectar objetos en tiempo real empleando cámaras con resoluciones VGA y obteniendo 95% de precisión para la vista frontal [9].

c. Máquinas de Vectores Soporte (SVM)

Las SVM fueron inspiradas en los resultados de la teoría de aprendizaje estadístico desarrollado por Vapnik en los 70's [16]. Este clasificador permite encontrar un hiperplano capaz de separar linealmente dos clases, proyectando el espacio de entrada original a un espacio de características altamente dimensional donde maximiza el margen entre clases.

Las SVM permiten estimar una función de clasificación óptima empleando datos de entrenamiento etiquetados como X_{tr} , de esta forma, la función f clasificará correctamente datos no vistos antes por el clasificador (datos de prueba). Considerando el caso más simple de clasificación binaria, asumimos que el conjunto X_{tr} es dado como en la ecuación (2):

$$(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n), \quad (2)$$

i.e. $X_{tr} = \{x_i, y_i\}_{i=1}^n$ donde $x_i \in R^d$ y $y_i \in R(+1, -1)$ corresponde a la etiqueta de clasificación de la muestra x_i . La función de clasificación se expresa en la ecuación (3):

$$y_i = \text{sign} \left(\sum_{i=1}^n \alpha_i y_i K(x_i \cdot x_j) + b \right), \quad (3)$$

donde $x = [x_1, x_2, \dots, x_n]$ son los datos de entrada. Un nuevo objeto x puede ser clasificado usando (3). El vector x_i es mostrado en la forma de producto punto. Las α'_i son multiplicadores de Lagrange y b es el bias obtenido al entrenar la SVM.

3. Metodología

a. Segmentación de la placa dentro de la imagen usando AdaBoost

El primer paso del método propuesto consiste en ubicar la placa dentro de la imagen extraída del video. Para realizar esta tarea, se entrenó el algoritmo AdaBoost con las



Fig. 5. Detección de placas en imágenes con vista frontal.



Fig. 6. Detección de placas en imágenes con vista lateral izquierda.

características de Haar provenientes de 300 imágenes positivas (en las que se marcó la placa) y 1000 imágenes negativas (donde no hay placas de vehículos), provenientes de videos con tres diferentes vistas: frontal, lateral izquierda y derecha. Asimismo, se definió una proporción para la placa de 40 pixeles de ancho y 18 pixeles de alto para reducir la complejidad del análisis.

En la Fig. 5 se puede notar que las placas detectadas presentan poca inclinación debido a que son frontales, sin embargo en los videos con vista lateral izquierda (Fig. 6) y derecha (Fig. 7) es necesario aplicar una corrección a la inclinación. En la Fig. 6 se muestra que la vista además de ser lateral izquierda, también es trasera, ofreciendo evidencias de la generalización de la posición de la placa dentro de la imagen y aunque existen algunos falsos positivos, estos serán eliminados en una etapa posterior por la SVM.

b. Corrección de la inclinación de la placa

Debido a que la imagen proviene de los videos grabados con cámaras IP con diferentes vistas, se debe aplicar una rotación a la placa para alinearla horizontalmente y facilitar la extracción de los caracteres. En la Fig. 8 se muestran las líneas detectadas, agrupadas en horizontales, verticales, sesgadas a la izquierda y sesgadas a la derecha de acuerdo a su pendiente; después una rotación es aplicada en sentido contrario al sesgo mayoritario.

c. Detección de falsos positivos en las placas usando SVM

Con el fin de mejorar la detección de las placas vehiculares, se usó el clasificador SVM, entrenado con un conjunto de muestras positivas formadas por los histogramas de las placas correctamente detectadas por AdaBoost, mientras que las muestras negativas las integraron

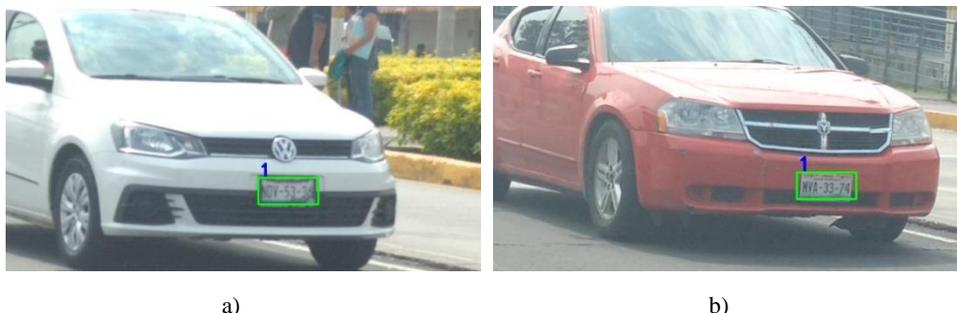
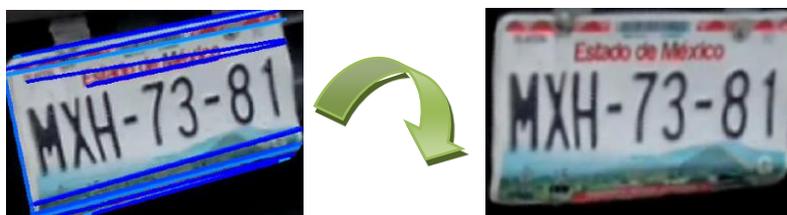


Fig. 7. Detección de placas en imágenes con vista lateral derecha.



a) Placa con sesgo a la izquierda

b) Placa con corrección del sesgo

Fig. 8. Corrección de la inclinación.

los histogramas de los falsos positivos. El entrenamiento con SVM empleó una función de base radial (RBF), definida en la ecuación (4):

$$K(x_i - x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0. \tag{4}$$

El parámetro C regula el punto medio entre error de entrenamiento y complejidad, mientras que γ es un parámetro del *kernel*. La obtención de buenos parámetros se logró usando una búsqueda en malla.

Para disminuir la complejidad del entrenamiento de la SVM, se usó el algoritmo Sequential Minimal Optimization (SMO) desarrollado por Platt [17].

d. Segmentación de los caracteres dentro de la placa usando Clustering

Una vez que la imagen de la placa ha sido extraída y alineada, se aplica un filtro de rango de color para dejar solo los pixeles negros y quitar las imágenes de fondo. Para facilitar la separación de los caracteres se quitan las filas con 80% de pixeles negros continuos que forman el marco superior e inferior como se muestra en la Fig. 9.

Después se buscan los pixeles negros continuos que formen un contorno cerrado y se agrupan de acuerdo a su altura y su posición en el eje “Y” para descartar los caracteres correspondientes al nombre del estado y manchas cerradas que pudieran confundirse como caracteres. El algoritmo se aplica máximo diez veces de forma iterativa, como se muestra

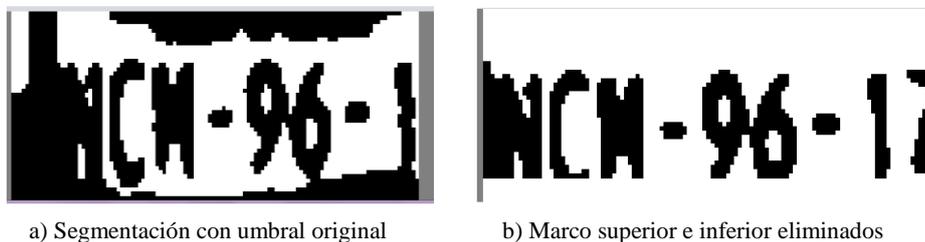


Fig. 9. Eliminación del marco superior e inferior de la placa.



Fig. 10. Segmentación aplicando clustering y rango de color.

en la Fig. 10, hasta encontrar el cluster con la mayor cantidad de contornos en forma de caracteres.

4. Resultados

En las pruebas realizadas se emplearon videos de cámaras IP colocadas para la investigación a un metro de altura en los semáforos del municipio de Texcoco debido a que estas no pueden enfocar el vehículo a más de 4 kilómetros por hora.

Para realizar los experimentos se grabaron nuevos videos y se prepararon los conjuntos de prueba de acuerdo a la vista de la cámara, destinando 149 fotos frontales, 42 para la vista lateral derecha y 50 para la izquierda. En la Fig. 11 se pueden ver algunos ejemplos de la detección de los caracteres para las placas con ruido por desenfoco, placas dobladas, con imagen de fondo, adornos en el marco, color de fondo y diferente longitud de caracteres.

En la prueba para ubicar la placa dentro de la imagen, el desempeño de AdaBoost con SVM fue de 95.97%, detectando 143 de 149 para la vista frontal; mientras que para la vista lateral derecha obtuvo una precisión de 85.71%, logrando ubicar 36 de 42 placas. En la prueba para ubicar los caracteres dentro de la imagen de la placa recortada, el desempeño

Tabla 1. Precisión para el método propuesto.

Conjunto de Datos	Segmentación de la placa		Segmentación de los caracteres
	Fotos	ACC	ACC
Frontal	149	0.95	0.83
Lateral izquierda	50	0.86	0.70
Lateral derecha	42	0.85	0.72



Fig. 11. Caracteres encontrados en la placa

de Clustering con el filtro por rango de color fue de 83.27% para la vista frontal, mientras que para la vista lateral derecha se obtuvo 72.53% como se muestra en la Tabla 1.

5. Conclusión

El análisis de los videos de seguridad es un campo de aplicación importante para la inteligencia artificial, tareas como la búsqueda de vehículos robados, aparcamiento y acceso a edificios requieren poder ubicar la placa dentro de la imagen antes de poder clasificar los

caracteres y obtener su número de serie. Las técnicas clásicas usan filtros para detectar los colores de la placa, su forma rectangular o una rejilla con las posiciones a partir de la norma del país y estado al que pertenece el vehículo, sin embargo, su precisión disminuye cuando la posición de la placa cambia debido a la altura del vehículo como camiones de carga, trailers y autobuses; cuando la vista ya no es frontal y las placas están muy desgastadas o tienen adornos en el marco.

Algunos autores se han centrado en mejorar la precisión de la clasificación de los caracteres, recortando las placas manualmente para poder entrenar los algoritmos, sin embargo en este artículo, se presentó un nuevo método que mejora la segmentación de las placas vehiculares; que a diferencia de otros reportados en la literatura, usa AdaBoost y las características de Haar para ubicar la placa a pesar de que pudiera localizarse en cualquier parte de la imagen, con diferente escala e inclinación.

La extracción de los caracteres también está generalizada ya que no depende de una ubicación fija, sino que agrupa los caracteres candidatos de acuerdo a su posición dentro de la placa, descartando los caracteres alrededor que pudieran corresponder al nombre del estado al que pertenece el vehículo, la marca o caracteres ajenos al diseño.

Otro problema es que en México cada estado usa imágenes de fondo conmemorativas, dificultando la segmentación, sin embargo, el método propuesto lo resuelve filtrando por rango de color hasta encontrar el cluster con todos los caracteres como se muestran en las figuras para las pruebas.

De acuerdo con los resultados, el método propuesto presenta una buena precisión, de 95.97% para segmentar las placas con vista frontal del automóvil, mientras que llega a 85% para la vista lateral derecha e izquierda, presentando un problema para clasificar aquellas con un sesgo de más de 45 grados. El método propuesto permite detectar las placas aun si son de otro tipo de vehículos como motocicletas o camiones de carga.

La segmentación de los caracteres fue 83.27% para la vista frontal, mientras que para la vista lateral derecha e izquierda obtuvo 70%, afectando la segmentación cuando las imágenes provienen de cámaras con resolución VGA y presentan mucho ruido.

Se propone comparar diversas técnicas para clasificar los caracteres de las placas, desarrollar un sistema distribuido para obtener los videos de diferentes cámaras IP HD ubicadas en lugares estratégicos como casetas, entradas de edificios, aeropuertos y semáforos para buscar autos con reporte de robo. También se requerirá usar algún algoritmo de optimización para encontrar los parámetros que mejoren la precisión de todo el sistema.

Referencias

1. Gu, Q., Yang, J., Kong, L., Cui, G.: Multi-scaled license plate detection based on the label-moveable maximal MSER clique. pp. 669–678 (2015)
2. Tedjojuwono, S.M.: Fast Performance Indonesian Automated License Plate Recognition Algorithm Using Interconnected Image Segmentation. In: Intan, R., Chi, CH., Palit, H., Santoso, L.(eds.) Intelligence in the Era of Big Data (ICSII) Communications in Computer and Information Science, 516, pp. 289–300, Springer, Berlin, Heidelberg (2015)
3. Xie, J., Zhou, H., Wu, X., Zhou, Y.: A Method of License Character Recognition Based on Fast Nearest Feature Line. In: Pan, Z., Cheok, A., Mueller, W., Zhang, M.(eds.) Transactions on Edutainment XI, (LNCS), 8971, pp. 52–60, Springer-Verlag, Berlin Heidelberg (2015)

4. Mohandes, M., Deriche, M., Ahmadi, H., Kousa, M., Balghonaim, A.: An Intelligent System for Vehicle Access Control using RFID and ALPR Technologies. Arab J SciEng, pp. 3521–3530, Springer, Berlin Heidelberg (2016)
5. Kluwak, K., Segen, J., Kulbacki, M., Drabik, A., Wojciechowski, K.: ALPR - Extension to Traditional Plate Recognition Methods. In: Nguyen, N.T., Trawiński, B., Fujita, H., Hong, TP.(eds.) Intelligent Information and Database Systems, ACIIDS 2016, Part II, Lecture Notes in Computer Science, 9622, pp. 755–764, Springer, Berlin, Heidelberg (2016)
6. On, C.K., Yao, T.K., Alfred, R., Ibrahim, A.A.A., Cheng, W., Guan, T. T.: A Comparison of BPNN, RBF, and ENN in Number Plate Recognition. In: Berry, M., Hj. Mohamed, A., Yap, B.(eds.) Soft Computing in Data Science, SCDS, Communications in Computer and Information Science, 652, pp. 37–47, Springer, Singapore (2016)
7. Liu, Y., Huang, H., Cao, J., Huang, T.: Convolutional neural networks-based intelligent recognition of Chinese license plates. Soft Comput. Springer-Verlag, Berlin Heidelberg (2017)
8. Fu, Q., Shen, Y., Guo, Z: License Plate Detection Using Deep Cascaded Convolutional Neural Networks in Complex Scenes. In: Liu, D., Xie, S., Li, Y., Zhao, D., El-Alfy ES., (eds.) Neural Information Processing, (ICONIP), Lecture Notes in Computer Science, 10635, Springer, Cham (2017)
9. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proc. Conf. on Computer Vision and Pattern Recognition (CVPR) Kauai, HI, pp. 511–518 (2001)
10. Rezaei, M., Ziaei-Nafchi, H., Morales, S.: Global Haar-Like Features: A New Extension of Classic Haar Features for Efficient Face Detection in Noisy Images. In: Klette, R., Rivera, M., Satoh, S. (eds.) Image and Video Technology, (PSIVT'13), Lecture Notes in Computer Science, 8333, Springer, Berlin, Heidelberg (2014)
11. Prasanna, D., Prabhakar, M.: An efficient human tracking system using Haar-like and hog feature extraction. Cluster Comput, pp. 1–8, Springer US (2018)
12. Elkerdawi, S.M., Sayed, R., ElHelw, M.: Real-Time Vehicle Detection and Tracking Using Haar-Like Features and Compressive Tracking. In: Armada, M., Sanfeliu, A., Ferre, M. (eds.) ROBOT2013, First Iberian Robotics Conference, Advances in Intelligent Systems and Computing, 252, Springer, Cham (2014)
13. Sotheeswaran, S., Ramanan, A.: A Coarse-to-Fine Strategy for Vehicle Logo Recognition from Frontal-View Car Images. Pattern Recognition and Image Analysis, vol. 28, pp. 142–154. Pleiades Publishing (2018)
14. Benabderrahmane, S.: Combining boosting machine learning and swarm intelligence for real time object detection and tracking: towards new meta-heuristics boosting classifiers. International Journal of Intelligent Robotics and Applications, vol. 1, pp. 410–428. Springer Singapore (2017)
15. Jiang, T., Cai, M., Zhang, Y., Zhao, X.: A Fast Video Vehicle Detection Approach Based on Improved Adaboost Classifier. In: Tan, Y., Takagi, H., Shi, Y., Niu, B. (eds.) Advances in Swarm Intelligence, ICSI, Lecture Notes in Computer Science, vol. 10386, Springer, Cham (2017)
16. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer-Verlag (1995)
17. Platt, J.C.: Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. Technical Report MSR-TR-98-14 (1998)