

Detección de lugares disponibles en un estacionamiento

Víctor Romero-Bautista¹, Aldrin Barreto-Flores¹,
Salvador E. Ayala-Raggi¹, Verónica E. Bautista-López²

¹ Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Electrónica,
México

² Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación,
México

victor.romerobau@alumno.buap.mx, {aldrin.barreto,
salvador.raggi, veronica.bautistalo}@correo.buap.mx

Resumen: Los sistemas de asistencia a estacionamientos basados en visión por computadora, cada vez son empleados con mayor frecuencia, para el mejoramiento del tránsito vehicular en el área urbana. Estos sistemas proporcionan información de la disponibilidad de un estacionamiento, además, contribuyen a tener una mejor organización en el mismo y a reducir el tiempo de búsqueda de un espacio disponible a los usuarios. El principal reto de estos sistemas se debe a la variación de iluminación que suele ocurrir durante el día, ya que puede presentarse una disminución de iluminación debido a efectos de sombra, o bien, incrementos a causa del sol, además de éstos, se pueden presentar otros inconvenientes, como variación de tonalidad en los autos, o la oclusión entre los mismos. En este trabajo se propone un método para la detección de lugares disponibles en un estacionamiento, en el cual se emplea un procedimiento para la ubicación semiautomática de regiones de interés (ROI). Se utilizó el modelo de intersección cortical (ICM) para la extracción de características, el análisis de componentes principales (PCA) para reducción de características y el perceptrón multicapa para la etapa de clasificación. El método propuesto presentó el 90% de efectividad en la detección, presentando baja susceptibilidad a los cambios de iluminación.

Palabras clave: Aprendizaje, detección, lugares de estacionamiento, firma, ROI.

Detection of Available Places in a Parking Lot

Abstract. Parking assistance systems based on computer vision are used more and more frequently to improve vehicular traffic in urban areas. These systems provide information on the availability of a parking lot, in addition, they contribute to having a better organization in it and to reducing the search time for an available space for users. The main challenge of these systems is due to the variation in lighting that usually occurs during the day, since there may be a decrease in lighting due to shadow effects, or increases due to the sun, in addition

to these, there may be other inconveniences, such as variation of tonality in the cars, or the occlusion between them. In this work, a method for the detection of available spaces in a parking lot is proposed, in which a procedure for the semi-automatic location of regions of interest (ROI) is used. Cortical Intersection Model (ICM) was used for feature extraction, Principal Component Analysis (PCA) for feature reduction, and multilayer perceptron for the classification stage. The proposed method presented 90% effectiveness in detection, presenting low susceptibility to lighting changes.

Keywords: Learning, detection, parking spots, signature, ROI.

1. Introducción

La detección de lugares disponibles en estacionamientos basados en visión por computadora, presentan diversas dificultades, para lograr una adecuada detección. Algunas de éstas, corresponden a los cambios de iluminación que ocurren a lo largo del día, efectos de sombra, diferentes tonalidades de color que pueden presentar los automóviles, occlusión entre los objetos de interés, además de otros. La detección de lugares disponibles en estacionamientos es comúnmente empleada en los sistemas de asistencia a estacionamientos, los cuales proporcionan información a los usuarios acerca de la disponibilidad de un estacionamiento, además, contribuyen a mejorar el flujo vehicular y la organización dentro del mismo.

Algunos trabajos propuestos en el estado del arte para la detección de lugares disponibles en estacionamientos, basan su operación en la detección de automóviles, como en [1], donde se propone un método para obtener las características de imágenes de autos, mediante el histograma de gradientes orientados (HOG), y para llevar a cabo la detección de éstos, emplea máquinas de vectores de soporte (SVM); el mismo enfoque se da en [2], donde se propone un método basado en la resta del fondo, y posteriormente la segmentación de la imagen resultante, usando la red neuronal pulso-acoplada (PCNN), para así llevar a cabo la detección de los autos.

Por otro lado, se encuentran los métodos que emplean la detección de bordes y esquinas, los cuales corresponden a las marcas que delimitan los espacios de estacionamiento, como es el caso de [3], el cual emplea el detector de esquinas de Harris.

También se encuentran los métodos que basan su operación en la ubicación de regiones de interés (ROI), éstos normalmente indican de forma manual las coordenadas de cada ROI en la imagen, como en [4], donde una vez generadas las ROI, aplica la resta del fondo a cada una de éstas, y para el reconocimiento de los objetos, se extrae el HOG, detector de esquinas mediante la transformada de características invariantes (SIFT) y métricas de los espacios de color YUV, HSV y YCrCb; en [5], de igual manera se aplica la ubicación de ROI señalando sus coordenadas manualmente, y para la detección de lugares disponibles, aplica un umbralado a cada ROI.

Métodos recientes para la detección de lugares disponibles en estacionamientos, llevan a cabo el uso de redes neuronales convolucionales (CNN), como en [6], donde se emplea la CNN para generar el *bounding boxing* en los automóviles; en [7] se emplea la CNN, para la extracción de características, y para clasificación se hace uso de SVM.

Estos trabajos presentan algunas ventajas para ciertos casos, pero también, son susceptibles a inconvenientes. Por ejemplo, en [1], [2], [4] y [7], se presenta un bajo rendimiento ante la presencia de sombras y variación de iluminación. Por otro lado, en [5], se tiene una respuesta mejor a cambios de iluminación, sin embargo, éste únicamente fue probado a escala. En [6], se presentó una baja efectividad, debido a que se utilizó un conjunto de datos limitado, y el trabajo presentado en [3], demanda que las líneas delimitadoras de los espacios de estacionamiento se encuentren uniformemente remarcadas.

En este trabajo se propone un método para la detección de lugares disponibles en un estacionamiento, en el cual se emplea un procedimiento para la ubicación semiautomática de ROI, por otro lado, para el reconocimiento de los objetos, se hace uso de la firma, que genera la red ICM, para la cual, se propone el valor de sus parámetros. Además, se lleva a cabo la reducción de características, mediante el PCA y para la clasificación de éstas, se hace uso de la red MLP.

El método propuesto se evaluó con un conjunto de 210 imágenes con cuatro tomas diferentes de topologías de estacionamiento, y se implementó en una Raspberry Pi 4. Se obtuvo una efectividad del 90% y un tiempo promedio de ejecución de 2 segundos.

2. Método propuesto

La propuesta del método consiste de los siguientes cuatro pasos: generación de regiones de interés (ROI), extracción de características, reducción de dimensionalidad y detección. A continuación, se presenta más a detalle cada uno de estos pasos.

2.1. Generación de regiones de interés (ROI)

En este primer paso, se generan las zonas donde se encuentran los lugares de estacionamiento. En algunos trabajos, como en [4] y [5], se lleva a cabo esta etapa, indicando las coordenadas de cada una dentro de la imagen, demandando tiempo importante para esta tarea.

Con el objetivo de disminuir el tiempo de ubicación de ROI, se optó por llevar a cabo la localización de éstas de manera semiautomática, de tal forma que, únicamente se requiera indicar la cantidad lugares de estacionamiento para ciertas zonas en la imagen.

De acuerdo con la distribución de los lugares de estacionamiento en el conjunto de imágenes utilizado para este trabajo, se identificaron dos zonas (superior e inferior), en las cuales se ubican los espacios de estacionamiento, éstas se presentan en la figura 1.

Partiendo de la distribución presentada en la figura 1, se diseñó un método para generar las ROI de forma semiautomática, en el cual únicamente se debe indicar la cantidad de lugares de estacionamiento que se encuentran en cada zona, considerando que la cámara de captura no cambie de posición.

Este método consiste, primeramente, en dividir la imagen en tres filas, como se presenta en el inciso (a) de la figura 2, las cuales se denominan: fila superior, intermedia e inferior. La altura de las filas superior e inferior corresponde al tamaño en alto del espacio de estacionamiento, mientras que la fila intermedia es el área restante y no es considerada.

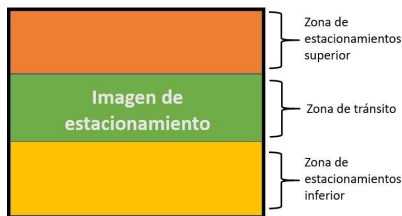


Fig. 1. Representación de distribución de lugares de estacionamiento en el conjunto de imágenes.

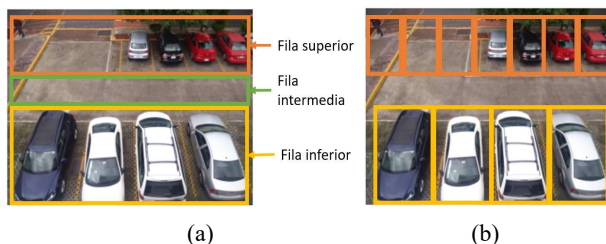


Fig. 2. Localización de regiones de interés; (a) generación de filas; (b) generación de recuadros (bounding boxing).

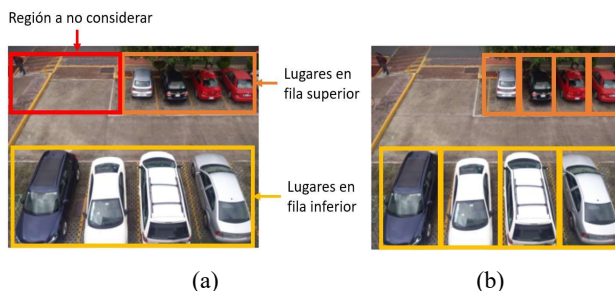


Fig. 3. Resultados de aplicar el método propuesto para la localización de ROI.

En seguida, las filas superior e inferior son divididas en columnas, de acuerdo al número de espacios de estacionamiento que se considere pueden ser contenidos dentro de cada fila, de tal manera que, cada columna pueda generar un recuadro que contenga un espacio de estacionamiento.

Como resultado, se presenta la imagen del inciso (b) de la figura 2, donde la fila inferior se ha dividido en cuatro columnas y la superior en siete, ya que, con esto, se logra contener un lugar de estacionamiento en cada recuadro generado.

El número de columnas a dividir es el parámetro que permite generar las ROI, y éste es establecido de acuerdo al tipo de toma con la que se esté trabajando.

Una vez hecho lo anterior, dependiendo de la topología del estacionamiento, se puede presentar una región que no se debe considerar, como se muestra en el inciso (a) de la figura 3, donde el recuadro en color rojo corresponde a la entrada del estacionamiento, por lo tanto, los recuadros generados dentro de esta región no deben ser tomados en cuenta.

Para esto, se estableció un parámetro, el cual indica si se deben ignorar algunos recuadros generados en cada fila de estacionamientos, que para el caso de la imagen del inciso (a) de la figura 3, se establece que los primeros tres recuadros (de izquierda a derecha), de la fila superior se deben saltar, obteniendo como resultado la imagen del inciso (b) de la figura 3, la cual muestra las regiones de interés generadas.

2.2. Extracción de características

En este paso, por cada ROI generada, se lleva a cabo la extracción de la firma ($G[n]$) producida por la red ICM, y a su vez, es almacenada en el vector de características a . La red ICM es un modelo simplificado de la red neuronal pulso-acoplada (PCNN) [8, 9], este tipo de red neuronal es ampliamente utilizada en el procesamiento digital de imágenes, como en: segmentación [10], reducción de ruido [11], detección de bordes [12], extracción de características [13], entre otras aplicaciones.

En las ecuaciones (1), (2) y (3), se presenta el modelo de red ICM, las cuales corresponden respectivamente a, el potencial interno (F), umbral dinámico (θ) y salida (Y), para una neurona i, j , donde S es la imagen de entrada y W la matriz de pesos sinápticos con tamaño k, l . Para el correcto funcionamiento de la ICM, se requiere del ajuste de los parámetros f, g y h .

La extracción de características con la red ICM, en general para una PCNN, se lleva a cabo mediante la obtención de la firma ($G[n]$), ésta cuantifica la cantidad de neuronas activadas por cada pulso generado durante su operación, y se expresa por la ecuación (4). Se llevaron a cabo una serie de pruebas con diferentes valores de parámetros, donde se concluyó que es posible obtener un patrón de firma adecuado, con una cantidad de 50 pulsos, empleando el valor de los parámetros: $f = 0,1$; $g = 0,8$; $h = 20$; y W como se presenta en la ecuación (5). La firma de las ROI, es generada tomando en cuenta los canales r, g y b, de la imagen:

$$F_{ij}[n] = fF_{ij}[n-1] + \sum_{kl} W_{ijkl} Y_{kl}[n-1] + S_{ij} \quad (1)$$

$$\theta_{ij}[n] = g\theta_{ij}[n-1] + hY_{ij}[n-1], \quad (2)$$

$$Y[n] = \begin{cases} 1 & \text{si } fF_{ij}[n] > \theta_{ij}[n], \\ 0 & \text{en caso contrario,} \end{cases} \quad (3)$$

$$G[n] = \sum_{i,j} Y_{i,j}[n], \quad (4)$$

$$W = \begin{pmatrix} 0,25 & 0,5 & 0,25 \\ 0,5 & 0 & 0,5 \\ 0,25 & 0,5 & 0,25 \end{pmatrix} \quad (5)$$

2.3. Reducción de dimensionalidad

Una vez que se ha generado el vector a , éste contendrá un total de 50 componentes, los cuales son normalizados a valores entre cero y uno, mediante la ecuación (6), generando así, el vector normalizado a_{norm} . Posteriormente se aplica la reducción de

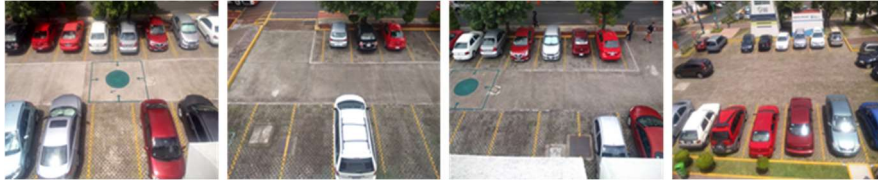


Fig. 4. Tomas correspondientes a las cuatro topologías de estacionamiento empleadas para la evaluación del método propuesto.



Fig. 5. Resultado de ubicación de regiones de interés.

dimensionalidad a dos componentes ($C1$ y $C2$), mediante el PCA [14], obteniendo el vector x , representado en la ecuación (7):

$$a_{norm} = \frac{a}{\max(a)}, \quad (6)$$

$$x = [C1, C2]. \quad (7)$$

2.4. Detección

Finalmente, para realizar la detección de los lugares disponibles, se lleva a cabo la clasificación del vector x , el cual es introducido a una red perceptrón multicapa (MLP) previamente entrenada [15], donde la salida de la red es enviada a una función de activación softmax [16], la cual es representada por la ecuación (8):

Algoritmo 1: Algoritmo propuesto para la detección espacios disponibles en un estacionamiento.

Input: I_{rgb}

Result: Imagen con ubicación de los espacios disponibles y ocupados I_r

```

1 Ubicar las regiones de interés (ROI) en  $I_{rgb}$ ;
2 for cada ROI do
3     Generar firma  $G[n] \rightarrow a$ ;
4     Generar  $a_{norm}$  Y reducir a dos componentes  $\rightarrow x$ ;
5     Clasificar  $x$  Mediante la red MLP  $\rightarrow \sigma$ ;
6     if  $\sigma[0, 1] > 80$  then
7         Marcar ROI como ocupado;
8     end
9     if  $\sigma[1, 0] > 80$  then
10        Marcar ROI como disponible;
11    end
12 end

```

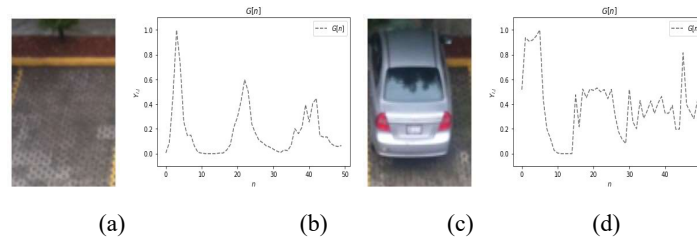


Fig. 6. Resultados de extracción de la firma $G[n]$.

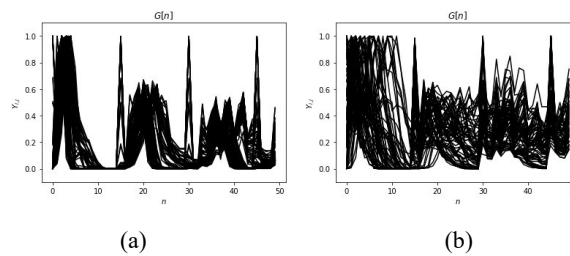


Fig. 7. Conjunto de firmas obtenidas de las imágenes de aprendizaje; (a) firmas generadas por espacios disponibles; (b) firmas generadas por espacios ocupados.

$$\sigma(y_k) = \frac{e^{y_k}}{\sum_{l=1}^c e^{y_l}}, k = 1, \dots, c. \quad (8)$$

donde σ corresponde a la salida de la red, dado en porcentaje de reconocimiento; y es la salida de una neurona en la capa de salida; k corresponde a la neurona actual y c a la cantidad de neuronas en la capa de salida.

Se estableció la etiqueta $[1,0]$ para identificar a la ROI que corresponda a espacios disponibles, y $[0,1]$ para ocupados. Donde la ROI será marcada como ocupada o disponible, si la salida de la red es mayor al 80%. En el algoritmo 2.1, se presentan el método propuesto para este trabajo.

3. Resultados

La intención de este trabajo es implementarlo en un estacionamiento, sin embargo, debido a la situación que se ha vivido por la pandemia, ésto no fue posible. Por lo anterior, se llevaron a cabo pruebas experimentales con 300 imágenes, las cuales contienen cuatro tomas de estacionamientos con topologías distintas (ver figura 4). Se utilizaron para la etapa de entrenamiento, un total de 90 imágenes, y las 210 restantes para evaluación.

A continuación, se presentan los resultados de la ubicación de ROI, extracción de características, reducción de componentes, etapa de aprendizaje y evaluación del método propuesto.

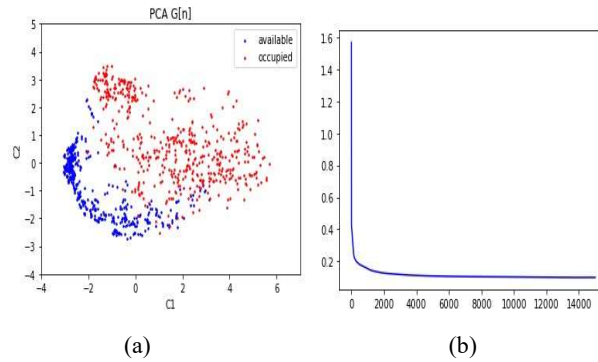


Fig. 8. Resultados de reducción de componentes y aprendizaje del clasificador MLP.

Tabla 1. Resultados obtenidos por el método propuesto y en trabajos del estado del arte.

Autores	Wahyono Tatulea Bibi Suhr Banerjee Acharya Propuesto						
	[1]	[4]	[5]	[3]	[6]	[7]	
Muestras de evaluación	2,656	106,192	200	134	2,000	504,139	2,009
Efectividad (%)	98.7	94.41	98	95	71.2	98.2	90
T. ejecución (segs)	-	-	-	-	-	2	2
Vel. de CPU (GHz)	-	-	-	-	-	2.5	1.5



Fig. 9. Resultados de la implementación del algoritmo propuesto en la tarjeta Raspberry pi 4.

3.1. Obtención de regiones de interés

En la figura 5, se muestra el resultado obtenido del método propuesto de ubicación de regiones de interés, para cada topología de estacionamiento, donde son resaltadas en color rojo las ROI generadas.

3.2. Extracción de firma ($G[n]$)

En la figura 6, se presentan los resultados de la extracción de la firma, donde la imagen del inciso (a), corresponde a un espacio disponible y la gráfica del inciso (b)

presenta su firma, por otro lado, la imagen del inciso (c) presenta un espacio ocupado por un automóvil, y la gráfica del inciso (d) indica su firma respectiva.

De acuerdo con las 90 imágenes utilizadas para la etapa de aprendizaje, se generaron un total de 940 muestras, de las cuales 498 pertenecen a espacios ocupados y 456 son de espacios disponibles. En las gráficas de la figura 7, se presenta el total de firmas recabadas de las imágenes de entrenamiento, donde el inciso (a) muestra el total de firmas obtenidas por imágenes de lugares disponibles, mientras que el inciso (b) presenta el total de firmas obtenidas por lugares ocupados.

3.3. Reducción de componentes y aprendizaje

Partiendo del conjunto de firmas generado por las imágenes de aprendizaje, se llevó a cabo la reducción de dimensionalidad de éstos, a dos componentes, mediante el método PCA, el resultado se presenta en la gráfica del inciso (a) de la figura 8.

Posteriormente, este conjunto de muestras se utilizó para el entrenamiento del clasificador MLP, el cual se configuró a una arquitectura piramidal, estableciendo dos capas ocultas, con once neuronas en la primera capa, nueve en la segunda y dos neuronas en la capa de salida, conectadas a una función de activación softmax. El entrenamiento del clasificador se llevó a cabo mediante el algoritmo *backpropagation*.

En la gráfica del inciso (b) de la figura 8, se presenta el decremento del error obtenido durante la etapa de aprendizaje del MLP.

3.4. Resultados de detección

Para la etapa de evaluación se utilizó un conjunto de 210 imágenes con cuatro topologías de estacionamiento diferentes. El algoritmo 2.1, fue implementado en lenguaje Python, el cual se ejecutó mediante una interfaz de usuario en la tarjeta Raspberry Pi 4, que cuenta con un procesador Quad Core Cortex-A72 con velocidad de 1.5 GHz y 4 GB de memoria RAM.

Se evaluó la velocidad de ejecución, así como, la efectividad, representada en la ecuación (9), donde FN corresponde a los falsos negativos, VP a los verdaderos positivos y M es el número de muestras:

$$efectividad = \frac{FN + VP}{M}. \quad (9)$$

En la tabla 1 se presentan los resultados obtenidos de la evaluación del método propuesto, en conjunto con los alcanzados por otros autores, en trabajos similares. Cabe mencionar que, en las métricas de tiempo de ejecución y velocidad de CPU, algunos autores no incluyeron esta información, por tal motivo, se ha dejado un espacio vacío en la tabla.

Además, en la figura 9, se muestran los resultados de la implementación del algoritmo en la tarjeta Raspberry Pi 4, donde son enmarcados en color verde los lugares disponibles y en rojo los ocupados.

Los resultados obtenidos, presentan una efectividad del 90%, el cual es 18% más, al alcanzado en [6], sin embargo, los trabajos [1, 5, 7], presentan aproximadamente una efectividad 8% mayor al obtenido por el método propuesto, mientras que, los trabajos [4, 3], presentan una efectividad del 5% más, al alcanzado.

Por otro lado, se obtuvo un promedio en velocidad de ejecución por imagen de 2 segundos, este resultado, es similar al generado en [7], el cual obtuvo un 8.2% más de efectividad que el algoritmo propuesto, sin embargo, éste utilizó 1 GHz más en velocidad de CPU.

4. Conclusiones y trabajo futuro

En este trabajo se ha presentado un método para la detección de lugares disponibles en un estacionamiento, el cual plantea la ubicación de ROI de manera semiautomática, considerando cuatro topologías de estacionamiento distintas, por otro lado, se proponen los valores de los parámetros de la red ICM, para emplearla en la extracción de características.

Bajo el método propuesto los resultados obtenidos presentan una efectividad del 90%, lo cual demuestra que es ligeramente robusto, presentando una baja sensibilidad a los cambios de iluminación, así como, a los efectos de sombra. Sin embargo, se presenta una diferencia de efectividad entre 5% y 8% menos, en otros trabajos del estado del arte.

Por otro lado, considerando la cantidad de imágenes empleadas, el trabajo propuesto, presentó resultados favorables, a pesar de contar un conjunto limitado de imágenes, en comparación con otros trabajos como lo son: [1, 4, 5, 6, 7].

Como trabajo futuro, se propone generar de manera automática las ROI a través del análisis de imágenes. Para mejorar la efectividad del algoritmo se propone, aumentar el conjunto de imágenes de estacionamientos, agregar una etapa de preprocesamiento en las ROI, para resaltar sus características.

Referencias

1. Wahyono, W., Hoang, V. D., Jo, K. H.: Multiscale car detection using oriented gradient feature and boosting machine. In: Nguyen, N., T., Trawinski, B., Fujita, H., Hong, T., P. (eds.) *Intelligent Information and Database Systems*, Springer, vol. 9621, pp. 731–740 (2016) doi: 10.1007/978-3-662-49381-6_70
2. Xu, Y., Wang, S., Li, X.: Vehicle video detection based on pulsed coupled neural network. In: *Proceedings of 7th International Congress on Image and Signal Processing*, pp. 731–735 (2014)
3. Suhr, J. K., Jung, H. G.: Fully-automatic recognition of various parking slot markings in around view monitor (avm) image sequences. In: *Proceedings of 15th International IEEE Conference on Intelligent Transportation Systems*, pp. 1294–1299 (2012)
4. Tătulea, P., Călin, F., Brad, R., Brâncovean, L., Greavu, M.: An image feature based method for parking lot occupancy. In: *Proceedings of Future Internet*, vol. 11, no. 8, pp. 169 (2019)
5. Bibi, N., Majid, M. N., Dawood, H., Guo, P.: Automatic parking space detection system. In: *Proceedings of Second international conference on multimedia and image processing (ICMIP)*, pp. 11–15 (2017)
6. Banerjee, S., Ashwin, T., Guddeti, R. M. R.: Automated parking system in smart campus using computer vision technique. In: *Proceedings of IEEE Region 10 Conference TENCON'19*, pp. 931–935 (2019)

7. Acharya, D., Yan, W., Khoshelham, K.: Real-time image-based parking occupancy detection using deep learning. In: Proceedings of the 5th Annual Conference of Research@Locate, vol. 2087, pp. 33–40 (2018)
8. Ma, Y., Zhan, K., Wang, Z.: Applications of pulse-coupled neural networks (2011)
9. Lindblad, T., Kinsler, J. M., Taylor, J. G.: Image processing using pulse-coupled neural networks. Springer, pp. 2 (2013)
10. Li, H., Guo, L., Yu, P., Chen, J., Tang, Y.: Image segmentation based on iterative self-organizing data clustering threshold of PCNN. In: 2nd International Conference on Cloud Computing and Internet of Things (CCIOT), pp. 73–77 (2016)
11. Rangel, E., Lavalle, M., Sossa, H.: Filtrado de ruido gaussiano mediante redes neuronales pulso-acopladas. *Computación y Sistemas*, vol. 21, pp. 381–395 (2017)
12. Shi, Z., Hu, J.: Image edge detection method based on a simplified PCNN model with anisotropic linking mechanism. In: Proceedings of 10th International Conference on Intelligent Systems Design and Applications, pp. 330–335 (2010)
13. Gu, X.: Feature extraction using unit-linking pulse coupled neural network and its applications. *Proceedings of Neural Processing Letters*, Springer, vol. 27, no. 1, pp. 25–41 (2008) doi: 10.1007/s11063-007-9057-6
14. Turk, M., Pentland, A.: Face recognition using eigenfaces. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 586–591 (1991)
15. Da-Silva, I. N., Spatti, D. H., Flauzino, R. A., Liboni, L. H. B.: Artificial neural networks. Springer Cham, vol. 39 (2017)
16. Marsland, S.: Machine learning. Second edition. Chapman and Hall/CRC (2015)