Vehicle Classification in Traffic Environments Using the Growing Neural Gas

Miguel A. Molina-Cabello(✉), Rafael Marcos Luque-Baena,
Ezequiel López-Rubio, Juan Miguel Ortiz-de-Lazcano-Lobato,
Enrique Domínguez, and José Muñoz Pérez

Department of Computer Languages and Computer Science,
University of Málaga, Bulevar Louis Pasteur, 35, 29071 Málaga, Spain
{miguelangel,rmluque,ezeqlr,jmortiz,enriqued,munozp}@lcc.uma.es
http://www.lcc.uma.es/~ezeqlr/index-en.html

Abstract. Traffic monitoring is one of the most popular applications of automated video surveillance. Classification of the vehicles into types is important in order to provide the human traffic controllers with updated information about the characteristics of the traffic flow, which facilitates their decision making process. In this work, a video surveillance system is proposed to carry out such classification. First of all, a feature extraction process is carried out to obtain the most significant features of the detected vehicles. After that, a set of Growing Neural Gas neural networks is employed to determine their types. A qualitative and quantitative assessment of the proposal is carried out on a set of benchmark traffic video sequences, with favorable results.

Keywords: Foreground detection · Background modeling · Probabilistic self-organising maps · Background features

1 Introduction

The field of traffic monitoring has generated great excitement in recent years within the intelligent transport systems community due to the increase of hardware development, the low cost sensor technologies and the improvement in the development and optimization of data processing algorithms. Specifically, the video detection and monitoring solutions for traffic applications can help to improve the performance in traffic management [3,10,13]. Thus, for example, if a high frequency of heavy vehicles is detected in one of the analyzed road sections, it is possible to redirect the traffic in a previous point with the aim of avoiding traffic congestion.

Automatic video surveillance systems can be divided into several phases [1,2]. A first step involves the detection of moving objects within the scene; a second stage performs monitoring tasks to associate the same vehicle detected in all frames of the sequence in which it appears; and finally a feature detection phase to extract relevant knowledge of the movement of these objects, their behavior and appearance.

I. Rojas et al. (Eds.): IWANN 2017, Part II, LNCS 10306, pp. 225–234, 2017.
DOI: 10.1007/978-3-319-59147-6_20

Each stage builds on the previous one, which implies that it is needed to have implemented the steps of detecting and tracking objects if it is intended to conduct an analysis of the detected vehicle. In this paper, some methods previously developed by our research team for object detection and tracking are combined with other techniques to yield a vehicle classification system. Specifically, a self-organizing neural network is applied to cluster the pixels in background and foreground layers in order to detect which pixels are in motion inside the scene [12]. Subsequently, a Kalman model for multiple objects is applied to determine the trajectory of each vehicle which appears in the scene [16].

Therefore, the aim of this work is to classify the detected vehicles in four categories: car, motorcycle, truck and van. A feature extraction process is required in order to obtain robust and discriminant characteristics which can differentiate correctly among the groups of vehicles. This analysis would help to manage and distribute the traffic more efficiently in the analyzed area. Other works in the literature have the same aim, although some of them apply different classification or clustering methods [4,8] or start from other methodologies associated to video surveillance systems [11]. In this case, the Growing Neural Gas model (GNG) is considered, since it has been used successfully in different classification problems, from novelty detection [5] to text classification [17], or even issues related to medicine or biology such as osteoporosis detection [15].

The rest of the paper is organized as follows: Sect. 2 presents our vehicle classification model, divided in an explanation of the base method (Subsect. 2.1) and its application to our classification problem (Subsect. 2.2). Section 3 shows several experimental results over several well-known public traffic surveillance sequences. Finally, Sect. 4 outlines the conclusions of the paper.

2 Model

This section describes a multi-class classification system consisting of a combination of several self-organizing models based on the growing neural gas approach.

2.1 Growing Neural Gas

The growing neural gas (GNG), [6], is an unsupervised neural network suited to vector quantization and clustering due to its capability of finding a topological structure which reflects the topology of the input distribution.

Let $X = \{\boldsymbol{x} \in \mathbb{R}^d\}$ be a set of M training patterns in a d-dimensional space. Let GNG be a growing neural gas model composed of N neurons. Initially, GNG starts placing $N = 2$ neurons randomly in the input space, which are supposed to be neighbors in the topological structure and, thus, connected by an edge.

Throughout the training phase, the topology is modified in two ways. First, the centroids of the clusters which the neurons represent $\boldsymbol{w}_i$ change, in order to adapt the self-organizing map that the GNG is creating to the input data distribution. For that purpose, the competitive learning rule is used. The competitive rule states that in a training iteration t only the neuron that best represents the currently presented training pattern $\boldsymbol{x}_t$, i.e. the neuron whose centroid is closest to it, can modify its own internal data, which in case of a GNG neural network are the neuron centroid $\boldsymbol{w}_{win}$ and a variable $error_{win}$ containing the accumulated error that would be obtained if a vector quantization was carried out.

$$win(t) = argmin_{1 \leq j \leq N}\{\|\boldsymbol{x}_t - \boldsymbol{w}_j(t-1)\|^2\} \tag{1}$$

$$\boldsymbol{w}_i(t) = \begin{cases} \boldsymbol{w}_i(t-1) + \eta_{win}(t)\ (\boldsymbol{x}_t - \boldsymbol{w}_i(t-1)) & \text{if } i = win(t) \\ \boldsymbol{w}_i(t-1) & \text{otherwise} \end{cases} \tag{2}$$

$$error_{win}(t) = error_{win}(t-1) + \|\boldsymbol{w}_{win}(t-1) - \boldsymbol{x}_t\|^2 \tag{3}$$

where $\|\cdot\|$ is the Euclidean norm.

In order to allow the map to self-organize and maintain the shape of the input distribution that it has captured, the neurons that are neighbors of the winner neuron are also able to slightly update their centroid position. In this case, a much smaller learning rate η_{neigh} is used for those neurons adjacent to the winner. Therefore, that new case is added to Eq. 2 and the final equation to model the centroid position update is

$$\boldsymbol{w}_i(t) = \boldsymbol{w}_i(t-1) + \begin{cases} \eta_{win}(t)\ (\boldsymbol{x}_t - \boldsymbol{w}_i(t-1)) & \text{if } i = win(t) \\ \eta_{neigh}(t)\ (\boldsymbol{x}_t - \boldsymbol{w}_i(t-1)) & \text{if } i \in Neighbors(win(t)) \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where $\eta_{win} : \mathbb{N} \longrightarrow [0,1]$ and $\eta_{neigh} : \mathbb{N} \longrightarrow [0,1]$ are two constant or monotone decreasing functions which satisfy $\forall t \in \mathbb{N} \quad \eta_{win}(t) > \eta_{neigh}(t)$.

Second, during the training phase new neurons are created periodically and useless neurons, also known as dead ones, are removed from the self-organizing-map. This is one of the advantages of growing neural networks such as the GNG, the number of neurons which are part of the model does not have to be fixed a priori by the user. However, a maximum number of neurons may be defined and it prevents GNG from creating too many neurons.

Every λ iterations a new neuron is inserted in the self-organizing map. The selection of the place to insert the neuron is based on the performance of the existing neurons. Those neurons with a high accumulated error perform poorly because the cluster which they represent are heterogeneous or contain many more elements than other clusters. Therefore, the new neuron centroid will be placed in the middle point of the edge that connect the neuron u with the largest error and the neuron v which is the neighbor of u accumulating the largest error.

$$\boldsymbol{w}_{new} = \frac{\boldsymbol{w}_u + \boldsymbol{w}_v}{2} \tag{5}$$

As *new* is between u and v the edge connecting both of them is replaced by two new edges: one from u to *new* and other one from *new* to v. The new neuron is assumed to represent some of the patterns which previously belonged to the clusters corresponding to u and v, thus the accumulated error of the three neurons are updated accordingly.

$$error_u(t) = \alpha \cdot error_u(t) \tag{6}$$

$$error_v(t) = \alpha \cdot error_v(t) \tag{7}$$

$$error_{new}(t) = error_u(t) \tag{8}$$

where $\alpha \in [0, 1]$ is considered the estimated fraction of accumulated error that is reduced after inserting the new neuron.

On the other hand, the edges between neurons have an associated variable *age* which is incremented as the training advances. Age of the edges connecting neurons which keep on winning the competition, i.e. those that are not dead, are refreshed and set to 0 (see Subsect. 2.1). Those edges whose age is greater than a given threshold a_{max} are removed, since the neurons they connect did not win the competition recently. If some neurons become isolated after removing edges, they are considered dead and are also removed from the map.

GNG Training Algorithm

1. The network is initialized by creating two randomly positioned nodes which are connected by an edge. Their accumulated errors are set to 0.
2. In time instant t, select randomly a vector $\boldsymbol{x}_t$ which has not been presented to the network previously, if possible.
3. Determine the winning neuron *win* using Eq. 1 and also the runner-up neuron *rup* with a reference vector $\boldsymbol{w}_{rup}$ such that $\|\boldsymbol{x}_t - \boldsymbol{w}_{rup}(t-1)\|^2$ is the second smallest, for all neurons N.
4. Update the accumulated error of the winning neuron by means of Eq. 3. Update the network topology using Eq. 4.
5. Age of all edges connecting *win* to its topological neighbors are incremented by one.
   If *win* and *rup* are connected by an edge, the age of that edge is set to 0. Otherwise, a new edge is created between them.
6. Edges with age larger than a_{max} are removed. If that results in some neurons without connections, those isolated neurons are also removed.
7. In case the maximum number of nodes has not been reached and $\{\exists j \in \mathbb{N} - \{0\} \mid t = j \cdot \lambda\}$ then a new neuron is created.
   (a) The neuron u with the largest error is determined as well as its neighbor neuron v with the largest error. The new neuron is inserted between them (see Eq. 5)
   (b) The edge between u and v is removed and two new edges are added, from *new* to u, and from *new* to v.
   (c) The accumulated errors for neurons u, v and *new* are updated following Eqs. 6, 7 and 8.
8. The accumulated errors are decreased by a factor β

$$\forall i \in [1..N] \quad error_i(t) = error_i(t) - \beta \cdot error_i(t) \tag{9}$$

9. If the finish requirements have not been meet, i.e. the maximum number of training steps has not been reached, go to step 2.

2.2 Classification System

The proposed multi-class classification system is based on a one-vs.-all approach, which involves training a GNG neural network for each one of the C classes to which the patterns can belong.

$$Classifier = \{GNG_i, \quad 1 \leq i \leq C\} \tag{10}$$

A given pattern $\boldsymbol{x} \in \mathbb{R}^d$ is assigned to the class i corresponding to the GNG which has the closest neuron to that pattern in the input space.

$$class(\boldsymbol{x}) = argmin_{1 \leq i \leq C}\{\|\boldsymbol{x} - \boldsymbol{w}^i_{winner}\|^2\} \tag{11}$$

where $\boldsymbol{w}^i_{winner}$ is the winner neuron (see Eq. 1) of GNG neural network GNG_i, which is trained to detect patterns that belong to class i.

The maximum number of neurons which form the different self-organizing maps has been set to the same value, $N = N_{max}$. That way none of the particular networks GNG_i is given an unfair advantage during the training phase.

3 Experimental Results

In this section we present the obtained results from our tests. The selected sequence to test our approach is the traffic video named U*S-101 Highway* which is available in the dataset of Next Generation Simulation (NGSIM) program, provided by the Federal Highway Administration (FHWA). This sequence presents several difficulties that should be dealt with, such as a perspective view from the outdoor scene, overlapping objects or occlusions.

The dataset, which contains information about the trajectories of several vehicles detected in a video sequence, presents small quantity of labeled vehicles and a large amount of them without any tag. Therefore, our proposal attempts to classify the vehicles that appear in the sequence into 4 possible classes: motorcycle, car, van and truck. An example of this different classes can be observed in Fig. 1, which shows 4 vehicles and its corresponding trajectories, one vehicle of each class.

Among the distinct features that can be extracted after segmenting an image, the selected ones for the tests have been the area, the perimeter, the width and the height of each object. Features like the object position in the scene are irrelevant when trying to identify its type and are not taken into consideration. Thus, for each segmented object O_i a set $\{\boldsymbol{x}^f_i \in \mathbb{R}^4 \mid f \in [1..MaxFrame]\}$ defines its trajectory during the video sequence. In order for the proposed classifier to work properly, a representative for each object is necessary. The chosen object description $\boldsymbol{x}^f_i$ is the one whose area coincides with the median of the area values in the members of O_i. Due to the robustness of the median statistic, those object descriptions corresponding to frames in which the segmentation failed or several objects overlap are never selected because they are far from the median value. Furthermore, only trajectories O_i with a minimum cardinality have been considered, in an attempt to avoid problems involving overlapping objects.

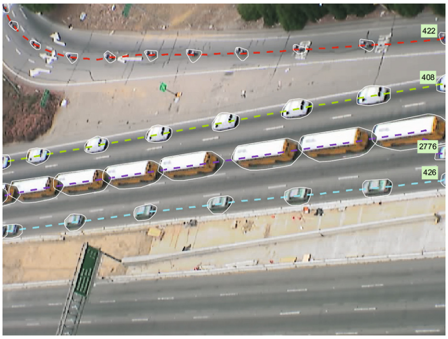


Fig. 1. Different vehicles and its trajectories detected by our approach. From top to bottom: vehicle 422 corresponds to the moto class, the trajectory 408 is a van, a truck can be observed with the vehicle 2776 and the vehicle 426 is a car.

The SOM methodology we have used is composed by 4 neural networks, one per class, that represents each considered vehicle type. Each network consists of 4 neurons. This number of neurons is low because the approach is based on an online learning strategy so that a desired real-time operation is attained.

To test the goodness of our proposal we have separated the labeled data in two random groups: the first one with the 90% of the data is used to train the model, and the second one with the remaining 10% of the data is managed to compare their vehicle type label with the proposed vehicle type result by the approach. The division of the groups has been carried out applying a stratified random selection. So each group has the same class object proportion.

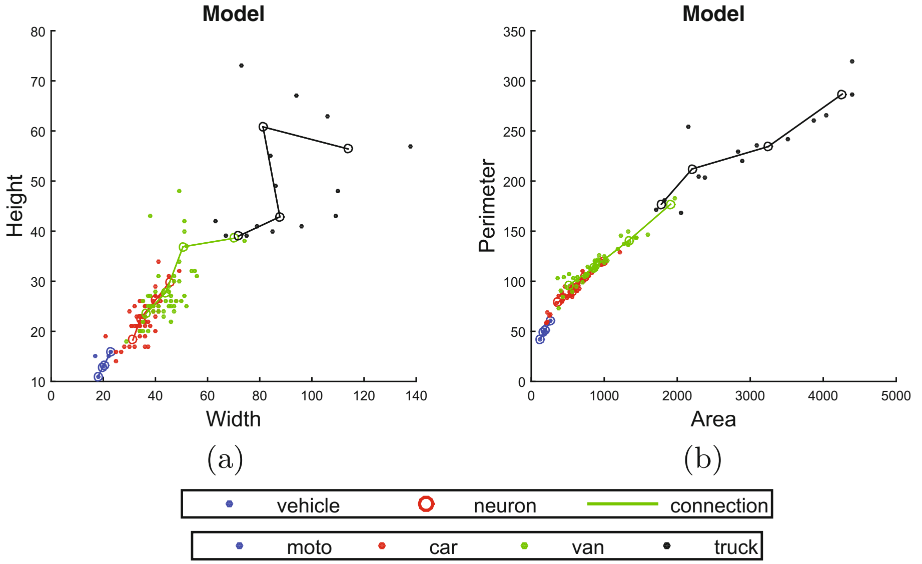
Thus, the process to obtain a classification of the objects is as follows. First of all, the labeled data is divided into 2 groups: training data and test data. Then the model is created and trained with the training data. Afterwards the model is proved with the test data. Finally, the unlabeled data are classified. This process has been carried out 10 times.

The distribution of one model with its neurons of each class model can be observed in Fig. 2: (a) shows the distribution organized by area and perimeter, and (b) exhibits the same distribution organized by width and height.

From a qualitative point of view, the produced classification results by the implemented model are shown in Fig. 3. The results organized by the area and the perimeter of each vehicle are presented in (a). In addition the same classification organized by width and height is visible in (b).

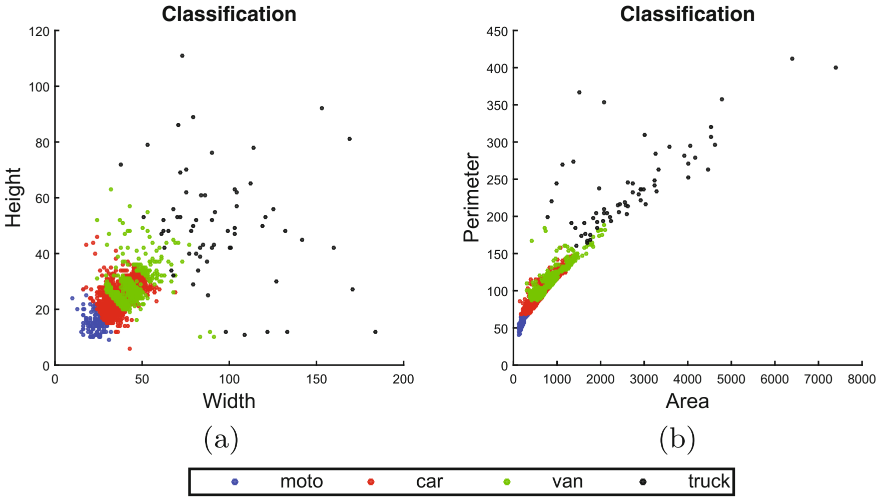
On the other hand, the quantitative results of the performance of the approach can be observed in Table 1. It shows some different well-known measures like the Accuracy or the Mean Square Error. Accuracy (Acc) is a value between 0 and 1, where higher is better. On the other hand, Mean Square Error (MSE) is a positive real number, where lower is better.

Let k be the observed object of the K existing objects, let $\mathbf{x}_k$ and $\mathbf{w}_k$ be the class of the object and the class suggested by the approach, respectively, where



(a) (b)

Fig. 2. Training data and the modelled neural network using 4 neurons per class. First column (a) shows data ordered by area and perimeter. Second column (b) presents the data sorted by width and height.



(a) (b)

Fig. 3. Classification produced by the model of the Fig. 2. First column (a) shows data ordered by area and perimeter. Second column (b) presents the data sorted by width and height.

$\mathbf{x_k}, \mathbf{w_k} \in \{1, 2, 3, 4\}$, corresponding $1 = moto, 2 = car, 3 = van$ and $4 = truck$. Furthermore, Let $\mathbf{q}_k = 1$ if the model hits the classification of the object k (so $\mathbf{x_k} = \mathbf{w_k}$) and $\mathbf{q}_k = 0$ if the model fails ($\mathbf{x_k} \neq \mathbf{w_k}$):

$$Acc = \frac{1}{K} \sum_{k=1}^{K} \mathbf{q}_k \tag{12}$$

$$MSE = \frac{1}{K} \sum_{k=1}^{K} (\mathbf{x}_k - \mathbf{w}_k)^2 \tag{13}$$

In addition we have considered other classification performance measures advocated in [14]. The Rand Index [9] measures the similarity between the clustering associated to the correct classification and the clustering associated to the predicted labels. It attains values in the interval $[0, 1]$ (higher is better), where 1 indicates a perfect classification. Hubert's Gamma Statistic [9] has values between -1 and 1 (higher is better), where 1 means a perfect correlation between the true and predicted labels. Finally, the Overall Cluster Entropy, the Overall Class Entropy and the Overall Entropy [7] measure the information about the correct labels which is contained in the predicted labels. These three measures have values in [0,1], where lower is better.

Table 1. Quantitative measures of the results. Each row is a measure and each column presents the median, the mean, the standard deviation, the best and the worst result for these measure, respectively.

Measure	Median	Mean	Best	Worst
Accuracy	0.7142	0.6785	0.8571	0.5000
Mean Square Error	0.2857	0.3214	0.1428	0.5000
Rand Index	0.6978	0.7000	0.8021	0.6043
Hubert's Gamma Statistic	0.2470	0.2737	0.5153	0.0739
Overall Cluster Entropy	0.5228	0.5290	0.3718	0.7305
Overall Class Entropy	0.5252	0.5294	0.3718	0.6364
Overall Entropy	0.5407	0.5292	0.3718	0.6834

The obtained results are influenced by the segmentation process (where vehicles are detected), and the tracking step (the trajectory of each vehicle is calculated). We have appreciated some overlapping problems, especially trucks because of its higher size, and this causes wrong classification classes. Other aspect that must be pointed out is the lower number of labeled objects, particularly motorcycles.

4 Conclusion

This paper proposes a novel approach based on the GNG (Growing Neural Gas) to determine the types of the vehicles appearing in traffic scenes. The detected vehicles are classified into four categories (cars, motorcycles, trucks and vans) by the system, based on a feature extraction process which provides the input data to carry out the classification.

The proposed neural architecture is composed by four GNGs, so that each GNG represents one of the considered vehicles types. Each vehicle is classified into the class associated to the GNG which best represents the features of the vehicle. Therefore, the proposed multiclass classification system is based on a one-vs-all approach, which involves the four GNGs. Due to the real-time operation requirement, an online learning strategy was used for the training of the four GNGs, whose number of neurons was also limited for the same reason.

The reported results of the simulation experiments in different video streams showed that the proposed approach achieved satisfactory classification results. The performance of the proposed approach depends on both the segmentation process and the tracking algorithm. Qualitative and quantitative evaluations of these results showed that the proposed approach performed at a high degree of vehicle detection rate, while the accuracy rate was satisfactory in the classification process.

Acknowledgments. This work is partially supported by the Ministry of Economy and Competitiveness of Spain under grant TIN2014-53465-R, project name Video surveillance by active search of anomalous events. It is also partially supported by the Autonomous Government of Andalusia (Spain) under projects TIC-6213, project name Development of Self-Organizing Neural Networks for Information Technologies; and TIC-657, project name Self-organizing systems and robust estimators for video surveillance. Finally, it is partially supported by the Autonomous Government of Extremadura (Spain) under the project IB13113. All of them include funds from the European Regional Development Fund (ERDF). The authors thankfully acknowledge the computer resources, technical expertise and assistance provided by the SCBI (Supercomputing and Bioinformatics) center of the University of Málaga. They also gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X GPU used for this research.

References

1. Baumann, A., Boltz, M., Ebling, J., Koenig, M., Loos, H., Merkel, M., Niem, W., Warzelhan, J., Yu, J.: A review and comparison of measures for automatic video surveillance systems. EURASIP J. Image Video Process. **2008**, 1–30 (2008)
2. Buch, N., Velastin, S., Orwell, J.: A review of computer vision techniques for the analysis of urban traffic. IEEE Trans. Intell. Transp. Syst. **12**(3), 920–939 (2011)
3. Cheng, H.Y., Hsu, S.H.: Intelligent highway traffic surveillance with self-diagnosis abilities. IEEE Trans. Intell. Transp. Syst. **12**(4), 1462–1472 (2011)
4. Crouzil, A., Khoudour, L., Valiere, P., Troung Cong, D.: Automatic vehicle counting system for traffic monitoring. J. Electron. Imag. **25**(5), 051207 (2016)
5. Fink, O., Zio, E., Weidmann, U.: Novelty detection by multivariate kernel density estimation and growing neural gas algorithm. Mech. Syst. Sig. Process. **5051**, 427–436 (2015)
6. Fritzke, B.: A growing neural gas network learns topologies. In: Advances in Neural Information Processing Systems 7, pp. 625–632. MIT Press, Cambridge (1995)

7. He, J., Tan, A.H., Tan, C.L., Sung, S.Y.: On quantitative evaluation of clustering systems. In: Wu, W., Xiong, H., Shekhar, S. (eds.) Clustering and Information Retrieval. Network Theory and Applications, vol. 11, pp. 105–133. Springer, Heidelberg (2004)
8. Huang, D.Y., Chen, C.H., Chen, T.Y., Hu, W.C., Lin, Y.L.: A vehicle flow counting system in rainy environment based on vehicle feature analysis. J. Inf. Hiding Multimedia Sig. Process. **7**(1), 101–114 (2016)
9. Jain, A.K., Dubes, R.C.: Algorithms for Clustering Data. Prentice-Hall Inc., Upper Saddle River (1988)
10. Kamijo, S., Matsushita, Y., Ikeuchi, K., Sakauchi, M.: Traffic monitoring and accident detection at intersections. IEEE Trans. Intell. Transp. Syst. **1**(2), 108–117 (2000)
11. Liang, M., Huang, X., Chen, C.H., Chen, X., Tokuta, A.: Counting and classification of highway vehicles by regression analysis. IEEE Trans. Intell. Transp. Syst. **16**(5), 2878–2888 (2015)
12. López-Rubio, E., Luque-Baena, R.M.: Stochastic approximation for background modelling. Comput. Vis. Image Underst. **115**(6), 735–749 (2011)
13. Luque-Baena, R.M., López-Rubio, E., Domínguez, E., Palomo, E.J., Jerez, J.M.: A self-organizing map to improve vehicle detection in flow monitoring systems. Soft Comput. **19**(9), 2499–2509 (2015)
14. Moschou, V., Ververidis, D., Kotropoulos, C.: Assessment of self-organizing map variants for clustering with application to redistribution of emotional speech patterns. Neurocomputing **71**(1–3), 147–156 (2007)
15. Podolak, I., Jastrzebski, S.: Density invariant detection of osteoporosis using growing neural gas. Adv. Intell. Syst. Comput. **226**, 629–638 (2013)
16. Rad, R., Jamzad, M.: Real time classification and tracking of multiple vehicles in highways. Pattern Recogn. Lett. **26**(10), 1597–1607 (2005)
17. Wang, X.J., Shen, H.: Improved growing learning vector quantification for text classification. Jisuanji Xuebao/Chin. J. Comput. **30**(8), 1277–1285 (2007)