IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received April 29, 2019, accepted May 28, 2019, date of publication June 4, 2019, date of current version June 17, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2920740

Moroccan Video Intelligent Transport System: Vehicle Type Classification Based on Three-Dimensional and Two-Dimensional Features

HATIM DERROUZ[1,2], (Member, IEEE), ABDERRAHIM ELBOUZIADY[1,2], (Member, IEEE), HAMD AIT ABDELALI[2], RACHID OULAD HAJ THAMI[1], SANAA EL FKIHI[1], AND FRANÇOIS BOURZEIX[2]
[1]ADMIR Laboratory, Rabat IT Center, IRDA Team, ENSIAS, Mohammed V University, Rabat 10000, Morocco
[2]Embedded System Department, MAScIR, Rabat 10100, Morocco

Corresponding author: Hatim Derrouz (h.derrouz@ieee.org)

This work was supported in part by the CNRST and in part by the MESRSFC, through the Development of an Integrated System for Traffic Management and Detection of Road Traffic Infractions Project.

ABSTRACT Vehicle type classification is a critical function in any intelligent transportation system (ITS). In this paper, we present a novel two-layer vehicle type classification framework based on the vehicle's 3D parameters and its local features. This framework is a part of the first Moroccan video intelligent transport system (MOVITS) that aims to control traffic and road code violations. In the first layer, the 3D features are extracted using the disparity map generated from stereo-images, and then, the width, height, and length of the vehicle are calculated based on the obtained list of 3D points. In the second layer, a gradient-based method is applied to extract the 2D features, and a dimensional reduction algorithm is performed to reduce its size. Both features are combined to construct the final feature vector that is used as an input for the classification. The Moroccan dataset and the BIT dataset were used to, respectively, validate the proposed framework and conduct a comparative study with the state-of-the-art algorithms. The experimental results demonstrate the efficiency of our approach against existing algorithms.

INDEX TERMS Computer vision, intelligent transport system, vehicle type classification, stereo vision, intelligent surveillance.

I. INTRODUCTION

Congestion is one of the most urging problems in traffic management all over the world, especially for emerging countries. Morocco is developing its first system to manage and control traffic in urban areas [1]. Moroccan video intelligent transport system (MoVITS) contains a set of functionalities including vehicle detection and tracking, vehicle recognition, speed estimation, anomaly detection, and traffic flux analysis. In the present work, we are going to describe the bloc responsible for vehicle recognition particularly vehicle type classification.

Vehicle type classification plays a very crucial role in intelligent transport system (ITS) since it can be used to identify the vehicle's type in anomalies case, counting vehicles by type, detecting road violations for some specific types, etc. It is a fine-grained classification. Unlike the typical classification problems based on classifying objects from different categories, the fine-grained classification job is to distinguish objects in the same category. This kind of classification represents a challenging task even for humans. This complexity is due not only to shared characteristics between some vehicle classes but also to the impact of the angle of view (Figure.1).

The challenge with the fine-grained vehicle classification is getting sufficient visual information and the adequate angle of view of the vehicle. Using a stereo-vision system could help solve these problems by enabling the extraction of three-dimensional information and by increasing the classification performance.

Stereo-vision has received much attention in the last few years in the area of transportation. It can be used for speed estimation [2], 3D vehicle reconstruction [3], Vehicle detection [4], etc.

In this paper, we combine the three-dimensional parameters with an appearance-based method to recognize the vehicle's type.

The associate editor coordinating the review of this manuscript and approving it for publication was Dong Wang.

A variety of experiments using a dataset collected from Moroccan urban areas and the well known BIT-dataset [5] are performed to validate the proposed framework.

The following section describes related work including background knowledge and prior research techniques. Our proposed system's architecture and methodology are presented in Sect. 3. The experiments and the obtained results are developed in Sect. 4. Our conclusions are drawn in the final section.

II. RELATED WORK

In the last decades, many researches have been proposed in the field of vehicle type classification. In general, we can distinguish between three approaches under which these methods are classified: Sensor-based approach, appearance-based approach, and model-based approach.

Very few studies have worked with the first approach which uses magnetic sensors and induction loop, In [6], for example, the authors use an algorithm based on the short-term variance sequence transformed from raw magnetic signal to detect vehicles and the Gradient Tree Boosting algorithm is employed to classify the detected vehicle into four categories. Experimental results on 4507 vehicles showed an accuracy rate of 99.8% for detection and 80.5% for identification.

The appearance-based approach which is frequently used for vehicle type classification gathers methods based on image and video processing under the perspective of extract visual feature from the image. In this context, the PCA-based method has been proposed by [7]. Authors process videos for segment individual vehicles, then they align vehicles along the same direction to normalize all vehicles in order to make them at the same scale; finally, they use two classifiers: the eigenvehicle and the PCA-svm.

In [8], authors classify vehicles into four classes: sedan, hatchback sedan, bus, and truck. Partial Gabor filter bank used to extract features and to classify vehicles. As reported by the authors, this framework has an optimal processing time and scored 95% of accuracy when applied to a dataset of 1100 images, including images with side view vehicles. Another method applied to side view vehicles was proposed by [9] and uses unsupervised learning discriminative features algorithm, in the context of matching non-overlapping cameras. A hybrid method used by [10] combines contour features and Speeded Up Robust Features (SURF) to extract the feature vector. They use hierarchical support vector machine (HSVM) to classify vehicles.

Some existing papers, focusing on the fine-grained issue, belong to the model-based approach. The authors of [11] present a visual-based method to estimate three-dimensional characteristics from images captured using a monocular camera. They extract vehicles from traffic image sequences and fit them with a set of predefined vehicle models after a shadow removal processing. Another method based on monocular images from a stationary camera is proposed by [12]. The authors tried to extract calibration parameters from ground truth images of the captured road. Based on calibration parameters, a set of features is extracted and used to construct the final feature vector, to be employed in the classification step. Hsieh et al. used the linearity features for vehicle representation [13]. They classified the vehicles based on their linearity representation. These methods are adapted to side view vehicles, and cannot perform well with rear and frontal view based-images.

On the other hand, fewer studies approached the frontal view case. Petrovic et al. extracted many features using a set of edge based methods such as Sobel edge response, edge orientation, direct normalized gradient, Harris corner, and locally normalized gradients [14]. A voting algorithm based on oriented-contour points applied to frontal view vehicles is used by [15] for vehicle type classification. They also used the SIFT method to extract other attributes, such as make and model, from the vehicles' logo. Pyramid histogram of oriented gradient (PHOG) and Gabor transform features are used by [16] to represent vehicles; a cascaded classifier schema was proposed to recognize the type of each processed vehicle. For the same purpose, [17] proposed a probabilistic parameter based on appearance and license plate position.

Deep learning techniques were also investigated for vehicle type classification purposes. Dong and his team used a convolutional neural network (CNN) to classify vehicles into six classes [5]. The sparse Laplacian filter learning (SLFL) was performed for feature learning, and the method managed to score 88% of accuracy on the BIT-Dataset.

Previous works focused majorly on particular cases and delivered high score when applied to specific views only. In general, Appearance methods are not sufficiently robust due to their sensibility to luminance changes and dependency on the quality of the image. In this paper, we propose a framework suitable for different views, that uses three-dimensional parameters combined with the gradient representation of the image.

III. CONTRIBUTION AND METHODOLOGY

A. CONTRIBUTION

The main contributions of this paper are:

- This is the most accurate framework for vehicle type classification that combines appearance-based feature vector of a 2D image with the 3D shape parameters extracted using stereo vision.
- This framework utilizes a robust feature vector, joining both local and global vehicle information.

B. METHODOLOGY

The present work aims to distinguish between six types of vehicles including sedan, SUV, bus, truck, minivan, and microbus. Our framework is capable of automatically identifying the type of vehicles, using the data served by a stereo vision system installed in an urban area. As shown in Figure 2, The 3D Vehicle Type Classification (3D_VTC) can be summarized in two steps: 3D feature extraction (3D FE) and 2D feature extraction & classification.

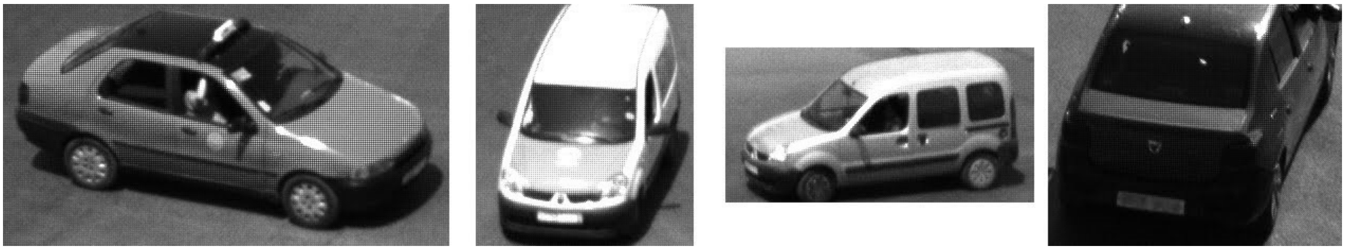

FIGURE 1. Vehicles from different angle of view.

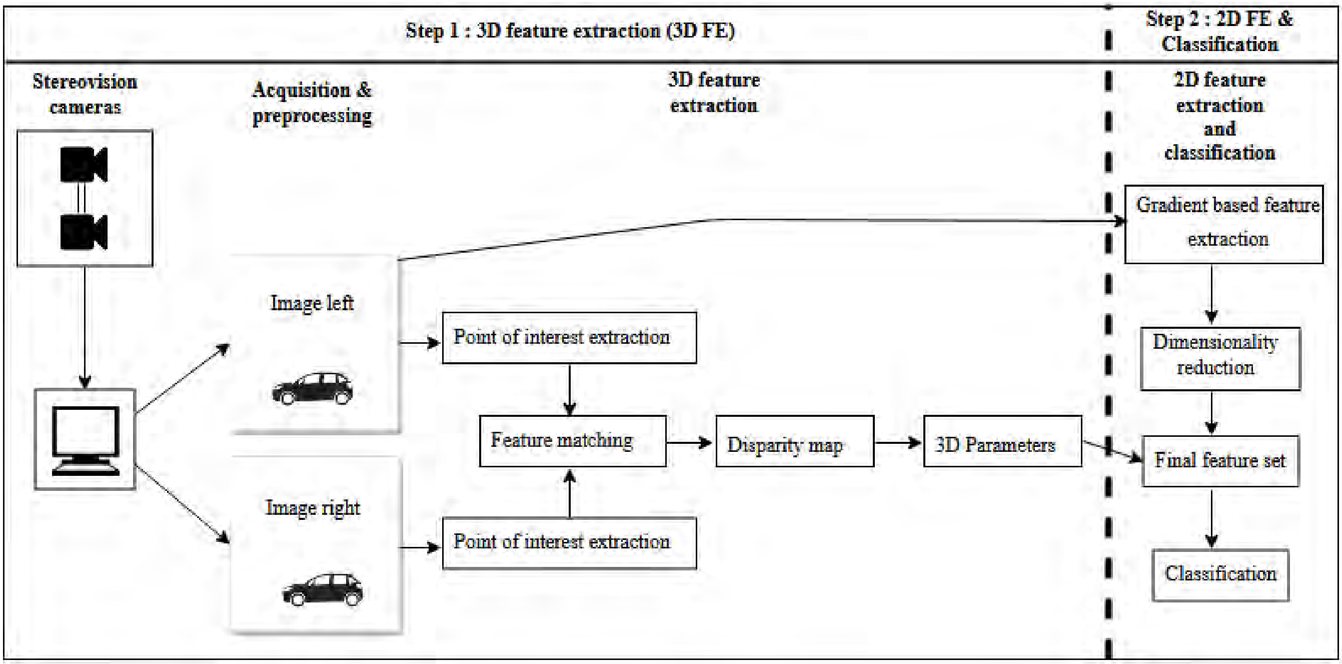

FIGURE 2. The flowchart of the proposed framework.

First of all, given that our system is based on stereo vision [18], it receives two images from both cameras left and right at the same time. After being acquired and preprocessed, the Yolo detector [19] is used to detect objects on the image, with the transfer learning technique [20]. The model will be tuned to only permit the vehicles detection. We apply features detector to identify key points (defined as the point of interest) in the vehicle from both images. The detected key points will be matched using brute force matching method and filtered to keep only shared key points from both inputs (left and right images). These matched key points will be used to generate the disparity map and afterward, obtain a set of 3D parameters (x, y, z). Accordingly, each vehicle will be assigned to a cube which width, height, and length are extracted using the min and the max values of the 3D parameters.

Next, in 2D feature extraction and classification step, the histogram of oriented gradient (HOG) [21] is applied to one of the input images to extract local information, and its size is reduced using principal component analysis (PCA) [22]. We then combine the 3D parameters with the resulting feature vector to construct the final feature set that represents the vehicle (see algorithm 4 line 4). The final feature set is fed to the five studied classifiers: support vector machine (SVM) [23], K-nearest neighbor (KNN) [24], Decision tree (DT) [25], Random forest (RF) [26], and Multilayer Perceptron (MLP) [27].

C. STEP.1: 3D FEATURE EXTRACTION (3D FE)

1) POINT OF INTEREST EXTRACTION

The aim of this part is to extract interesting and highly discriminant features from the input image such as blobs, edges, corner, etc. The scale-normalized determinant of hessian is used to detect the point of interest from the input image. Therefore, a sliding window (eq. 1) is used to parse the entire image and simultaneously, the determinant of Hessian is applied to extract the local changes around the concerned point. Consequently, the points with the highest determinant value (max(determinant)), as found by Eq. 2, are considered as the points of interest.

Algorithm 1 summarizes the process applied to detect the point of interest from the input image. First, we calculate the integral image using eq 1. So, it has an input image left and image right after being acquired, and as output, it gives a list of key points from both images.

Algorithm 1: Point of Interest Detection

Input : Image_left, Image_right
Output: Image_left_keypoints, Image_right_keypoints

if *Image_left & Image_right are not empty* **then**
  $Uleft \leftarrow INTEGRAL\_IMAGE(Image\_left)$
  $Uright \leftarrow INTEGRAL\_IMAGE(Image\_right)$
  $Image\_left\_keypoints \leftarrow HESSIAN\_DETERMINANT(Uleft)$
  $Image\_right\_keypoints \leftarrow HESSIAN\_DETERMINANT(Uright)$
end

The Convolution of a discrete function $f : \mathbb{Z}^2 \rightarrow \mathbb{R}$ with a discrete filter $g : \Omega \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$ with finite support Ω is defined as:

$$\forall(x, y) \in \mathbb{Z}^2, (f * g)(x, y) := \sum_{(i,j)\in\Omega} f(x - i, y - i)g(i, j), \quad (1)$$

where x and y are the coordinate of a pixel in the image.

Let w be the window of the input image so that the determinant of Hessian DoH(w) is calculated as follow:

$$DoH^L(w) := \frac{1}{L^4}(D_L^{xx} w.D_L^{yy} w - (kD_L^{xy} w)^2), \quad (2)$$

where D_L^{xx},D_L^{yy}, and D_L^{xy} represent the second order box filter at scale L, and k is a constant weighting factor equal 0.912 referred to [28].

2) FEATURE MATCHING & DISPARITY MAP

In the initial stage of the process, we compute the disparity map from stereo images, for that we need to match the extracted points on the left and the right image using brute force matching.

Fig. 3 illustrates the process of extracting the 3D location of a given point P. Let o be an object viewed by two cameras positioned in the same orientation and separated by a baseline distance (distance from two cameras in stereo vision). The horizontal distance between the projection of the object in both cameras is called disparity, and it is defined by disp in Eq 3 where x_L and x_R are the projected X-coordinate in the stereo vision cameras. Since both cameras are located on the same plane, Y-coordinates are the same for the two images ($y_L = y_R$).

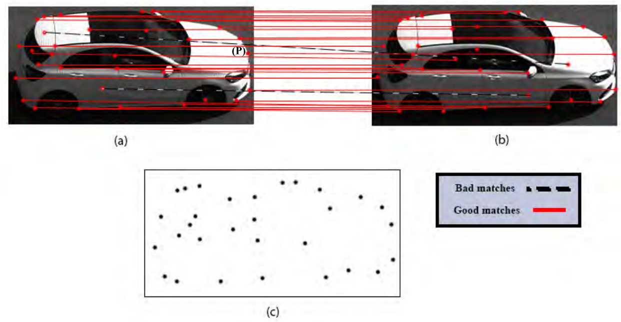

FIGURE 3. Example of disparity map for a vehicle.(a) the image from camera left, (b) the image from camera right, and (c) the generated map.

Algorithm 2 summarizes the steps where we examine each point of interest defined by key point detected and match between them for both images with brute force matcher. Then, we use a filter to leave only good matches. Finally, we calculate Z-coordinates based on the disparity map and triangulation method as described in the next subsection. Thus, the algorithm gives as output a list of the 3D point.

3) 3D PARAMETERS

To extract the 3D point, we use a method called Triangulation [29]. This method consisted to determine depth from disparity disp. Let f be the focal length, b the baseline and (x_L,y_L), (x_R,y_R) are the corresponding points for the left and right image respectively. The following equations can express the location of the 3D point:

$$Z = \frac{f * b}{(x_L - x_R)} = \frac{f * b}{disp} \quad (3)$$

$$X = \frac{x_L * z}{f} \; or \; \frac{b + x_R * z}{f} \quad (4)$$

$$Y = \frac{y_L * z}{f} \; or \; \frac{y_R * z}{f} \quad (5)$$

In Algorithm 3, we describe how to compute real-world dimensions of the vehicle step by step. So, it has as input the

Algorithm 2: Key point Matching & Compute Disparity map

Input : Image_left_keypoints, Image_right_keypoints
Output: List3DPoint

$matches \leftarrow Brute\_force\_matcher(Image\_left\_keypoints, Image\_right\_keypoints)$
$matches \leftarrow filtre(Matches, max\_dist, max\_slope, min\_slope)$
for $m \in matches \left\{(keypoint\_left_1, keypoint\_right_1), (keypoint\_left_2, keypoint\_right_2), \text{âĂę}, (keypoint\_left_n, keypoint\_right_n)\right\}$ **do**
  $disp \leftarrow keypoint\_left_i.x - keypoint\_right_i.x$
  $List3DPoint.add(Triangulation(m, disp))$
end

Algorithm 3: Compute Real World Dimensions of the Vehicle

Input : List3DPoint
Output: Width, Height, Length

Width = *Abs*(*Max*(*List*3*DPoint*.*x*) − *Min*(*List*3*DPoint*.*x*))
Height = *Abs*(*Max*(*List*3*DPoint*.*y*) − *Min*(*List*3*DPoint*.*y*))
Length = *Abs*(*Max*(*List*3*DPoint*.*z*) − *Min*(*List*3*DPoint*.*z*))

list of 3D point based on disparity map. Finally, The width, height, and length of the vehicle's cube are calculated.

D. STEP.2: 2D FE & CLASSIFICATION

1) GRADIENT BASED FEATURE EXTRACTION

Feature description is considered one of the most used methods to describe images for classification. Therefore, a various method has been proposed in this field, in the present work gradient information is used to extract the feature vector based on the orientation of the pixel's gradient.

To avoid luminosity problems, The histogram of oriented gradient (HOG) has proven the best performance [21]. For that, the hog method is used in the proposed framework. The following steps are applied to the input image:

a. Compute gradient for the x and y axis (G_x and G_y resp.) of the input image I_f:

$$G_x = I_f * [-1, 0, 1] \tag{6}$$
$$G_y = I_f * [-1, 0, 1]^T \tag{7}$$

b. Compute the magnitude of each gradient:

$$G = \sqrt{G_x^2 + G_y^2} \tag{8}$$

c. Compute gradient orientation θ:

$$\theta = arctan\frac{G_x}{G_y} \tag{9}$$

Next, the input image I_f will be divided into 8x8 cells and for each cell, a histogram of gradients will be calculated and represented using a 9-bin histogram in order to construct the final feature vector.

2) STEP.2: DIMENSIONALITY REDUCTION

To minimize the processing time, Principal Component Analysis (PCA) [22] has been performed to reduce the dimension of the HOG feature vector. The problem is to define an optimal components number that can resume the useful information in a small number of regenerated features. Under this scoop, our classifier loops over different numbers of regenerated features whereas plotting classification accuracy for each case in order to select the most accurate dimension.

3) STEP.2: FINAL FEATURE SET CONSTRUCTION & CLASSIFICATION

As soon as the reduced gradient-based feature set and three-dimensional parameters are ready, we combine them into one feature vector.

In this step, the classification has been made using five classifiers: The Support vector machine (SVM), Multilayer Perceptron (MLP), Decision tree (DT), K-nearest neighbor (KNN), and Random forest (RF).

Algorithm 4: Classification of Vehicles

Input : Image_left, Width, Height, Length
Output: Vehicle_type

while *Image_left & Width & Height & Length are not empty* **do**
  Feature_vector ← *Compute_HOG*(*Image_left*)
  Feature_vector ← *PCA*(*Feature_vector*)
  Feature_vector ← *Feature_vector* + *Width* + *Height* + *Length*
  Vehicle_type ← *CLASSIFICATION*(*Feature_vector*)
end

Algorithm 4 summarizes the final stage of our framework. It defines the main functions used to compute hog, reduce dimension (PCA), calculate the final feature vector, and doing classification.

IV. EXPERIMENTAL RESULTS

A. DATASETS, SETTINGS, AND PREPROCESSING

1) DATASETS

In the present work, a complex and challenging vehicle type dataset has been collected using a stereo vision system in the Moroccan urban areas. This Dataset contains videos recorded from a particular case in Moroccan roads; it contains challenging scenarios such as occlusion, shadows, and different time of the day. Figure.4 shows a sample of images from the collected dataset. MoVITS-Dataset contains more than 60 videos and 75,230 images annotated with a bounding box (Ground Truth) with 5 megapixels, 15 frames per second at a different time and places. Since 3D_VTC proposed for Moroccan scenarios, we have performed tests on this real dataset. In order to test different scenarios, we perform a cross-validation technique with a random selection method.

2) SETTINGS

To construct a stereo-vision system, we need to align two identical cameras fixed on a stereo bar with a baseline that separates them. Both cameras should have parallel optical axes. In order to configure this system, we should apply the following steps:

1) Synchronization: In this step, we configure both cameras to capture an image each exactly at the same time. For this, we have used a hardware trigger to activate both cameras with a signal.

FIGURE 4. Images from MoVITS-dataset.

TABLE 1. Corresponding number of each class name.

Class name	Corresponding number
Bus	0
SUV	1
Sedan	2
Minivan	3
Microbus	4
Truck	5

2) Calibration: This step is very critical before image acquisition process. Since a bad calibration will influence the system's accuracy. Two calibration method may be used to calibrate this system such as internal calibration which adjusts internal camera parameters, e.g.(aperture size, focal distance, etc.) to remove image distortions and external calibration that set two parallel optical axes of both cameras by adjusting camera baseline length, camera orientation, etc.

3) PREPROCESSING

Three-dimensional vehicle type classification (3D_VTC) is based on the pre-detected vehicle. For that, before performing classification, Yolo object detection is applied to the input image in order to extract a region that contains vehicles. Yolo gives as an output a set of rectangles that contains multiple objects, we leave only the vehicles, and we remove the other objects.

B. RESULTS

3D_VTC tested on our dataset that contains stereo images. Our approach achieves 95% overall accuracy. In Table 1 we define the correspondence between the numbers and the class names. To visualize the impact of adding three-dimensional parameters to the feature vector, we generate the confusion matrix for our data with three-dimensional parameters (see Figure 6) and without it (see Figure 5). The projected results show the effect of combining both parameters. Distinctly, for "Sedan" and "Microbus", when we had added the three-dimensional parameters, the model was able to classify them well.

Overall, it seems that most misclassifications are between "SUV" and "Sedan". It could be explained by the fact that they have some similar characteristics in the visual context, and also because they belong to the same category (light vehicles). The only visual difference we can distinguish between these two categories is the size of the vehicle. Our proposed framework can correctly classify vehicle types in the most challenging situations, like a different angle of views and different time of the day. The aims of combining three-dimensional parameters with visual descriptor are to describe the vehicle as well as possible. So, the use of global information (3D parameters) can provide discriminative features for the classifier. To classify vehicles five classification methods have been investigated, every classifier used with its parameters.

Neural network classifiers are widely used to classify the object, but they need significant data to give good results, for that we used only Multilayer perceptron, that represent a small neural network with three layers such as input layer, hidden layer, and the output layer. A set of parameters are tested in order to obtain the best accuracy.

SVM with RBF kernel has been used to classify the obtained data, in order to find the most accurate hyper-plane that separate each class. Multiple parameters are tested with various kernel type. The choice of the RBF kernel function made because it is suitable with this kind of classification problems [30].

Another classifier, such as K-nearest neighbor, Decision Tree, and Random forest has been used, but none of them has shown excellent results.

In Table 3, we give a comparative study between the five classifiers. As shown in Table 3, we compare the five classifiers used in this experience. A set of classical metrics such as precision, recall, and f-measure is used to evaluate 3D_VTC framework. SVM showed its superiority compared with other classifiers. The well-known SVM works well with less data (Our case). The classification is made for six classes with different characteristics adding the three-dimensional parameters gives the data much separate for that SVM gives good results. SVM aims to find the optimal hyperplane that separate class, and with the combination between three-dimensional parameters and the gradient-based feature vector, classes have become well separated.

The ROC (Receiver Operating Characteristics) curve is extracted, in order to evaluate the impact of adding three-dimensional parameters to the feature vector. In the Figure 8 we project the ROC curve for SVM classifier. Most of the six classes are highly accurate; this result is confirmed earlier by precision and recall of each class with SVM classifier.

Another method has been tested to the dataset in the interest of comparing its accuracy with the proposed framework. In this context, deep learning has been investigated using Transfer learning method with VGG16 [31], the accuracy does not exceed 88% and for ResNet [32] the accuracy archives 85,4% for overall classification in the best case. None of them gives good results due to the insufficient data for training as reported by the loss graph in Figure 7.

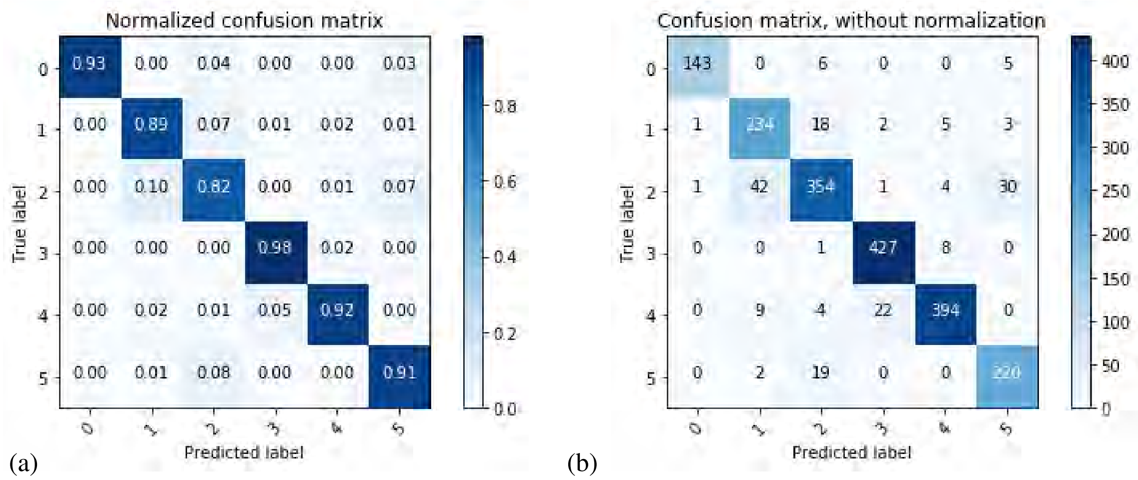

FIGURE 5. Normalized confusion matrix (a) and non-normalized (b) without three dimensional parameters using SVM classifier.

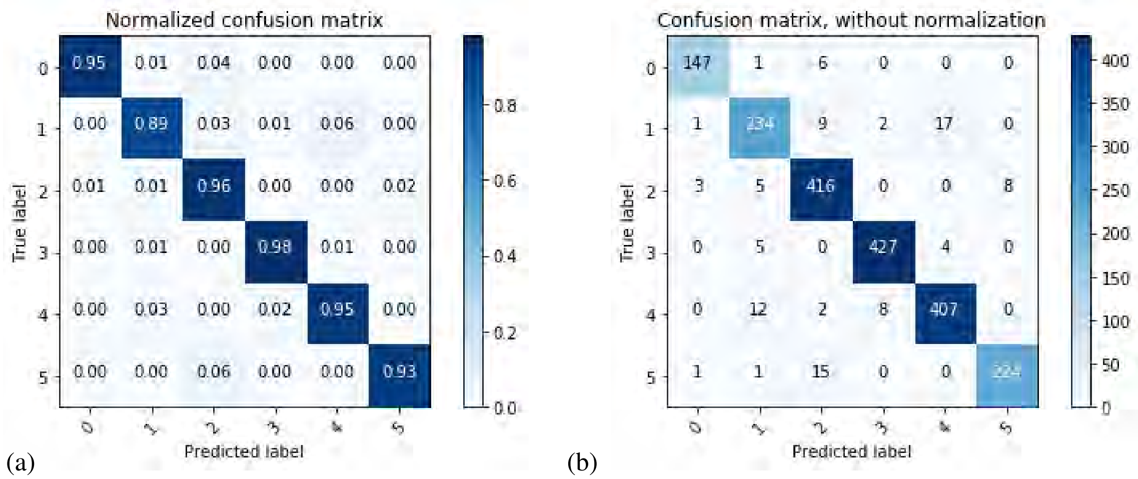

FIGURE 6. Normalized confusion matrix (a) and non-normalized (b) with three-dimensional parameters using SVM classifier.

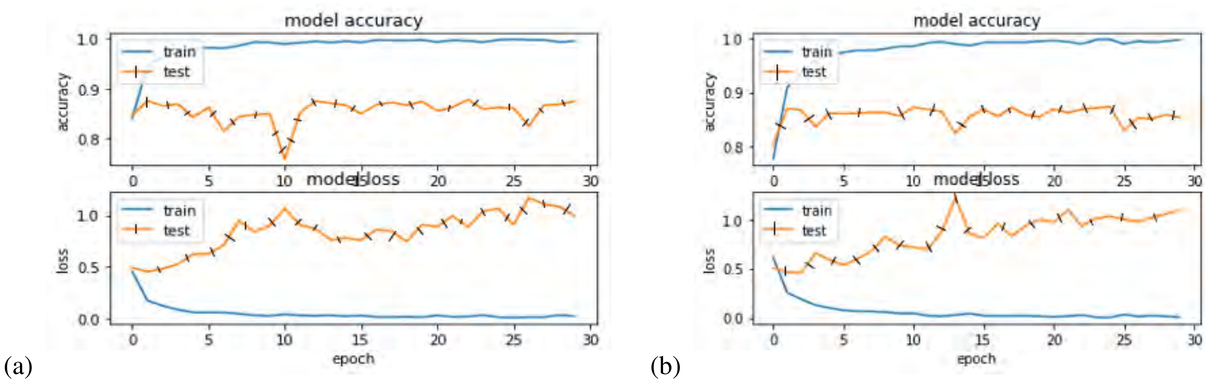

FIGURE 7. Vehicle type classification fine-tuning using VGG16 (a) and ResNet (b).

C. PERFORMANCE IN CHALLENGING CONDITIONS

The 3D_VTC framework performs well even in challenging scenarios: non-frontal views images and vehicles under occlusion as illustrated in Fig. 9. The invariant to such challenging conditions could be attributed to the impact of adding the three-dimensional parameters to the feature vector which combines local and global features.

D. COMPARISON RESULTS

To validate our work, BIT-vehicle Dataset [5] is used (see Figure.10). This dataset contains 9,850 vehicle images

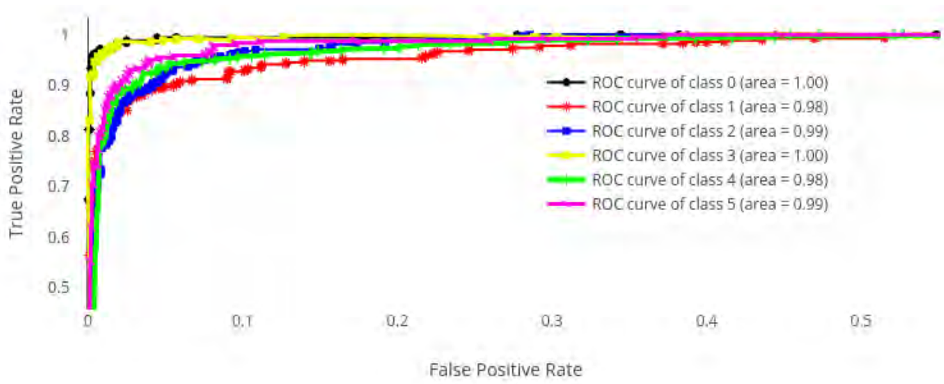

FIGURE 8. ROC curve of vehicle type classification using three dimensional parameters and gradient based features.

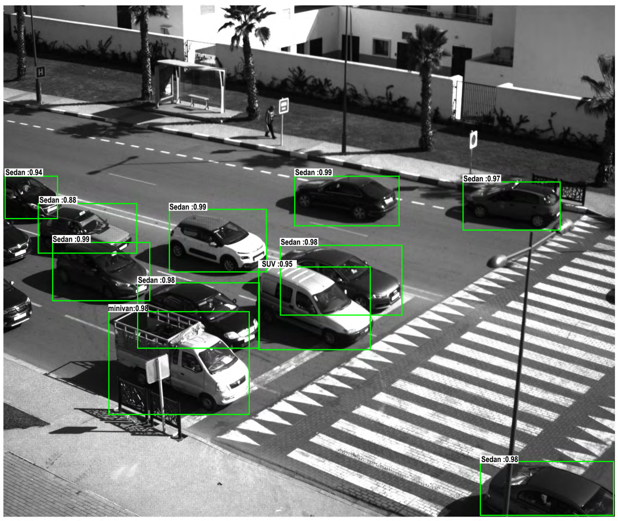

FIGURE 9. Vehicle type classification in challenging cases.

TABLE 2. FHWA vehicle type characteristics.

Vehicle Type	Width (m)	Height (m)	Length (m)
Bus	2.55	3.7	12.95
SUV	1.75	2	4.5
Sedan	1.75	1.45	4.5
Minivan	2.55	3.7	5.4
Microbus	2	2.7	7.8
Truck	2.5	3.4	7.8

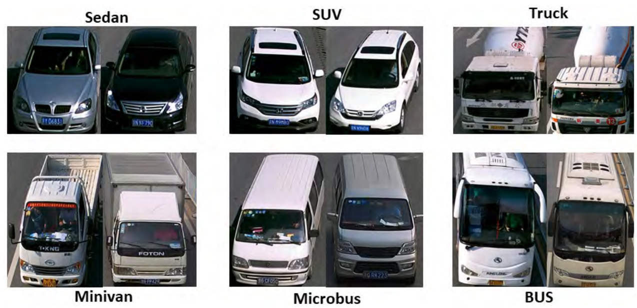

FIGURE 10. Images from BIT-dataset [5].

divided into six vehicle type classes: Truck, Van, SUV, Sedan, Microbus, and Bus.

The 3D_VTC framework is tested on the BIT-Dataset in order to compare our method with the state of the art methods. The experiments were done on daylight and night-light images. Thus, we have added the three-dimensional parameters manually using the Federal Highway Administration (FHWA) classification [33] (Table 2).

Our method achieves 95.2% in overall classification which is 93.1% on nightlight image and 97.3% on daylight images better than the state of the art. The results are demonstrated in Table 3. The reason behind the obtained results is the global

TABLE 3. Classification metrics using five classifiers for vehicle type classification.

	Classes	Bus	Microbus	Minivan	Truck	SUV	Sedan	Avg.
	Support	154	263	432	436	429	241	1955
SVM	Precision(%)	97	**91**	93	**98**	**95**	**97**	**95**
	Recall (%)	**95**	**89**	**96**	**98**	**95**	**93**	**95**
	F-measure (%)	**96**	**90**	**95**	**98**	**95**	**95**	**95**
MLP	Precision(%)	96	88	90	96	90	90	92
	Recall (%)	95	84	89	95	91	91	91
	F-measure (%)	96	86	89	96	91	90	91
KNN	Precision(%)	**99**	89	**98**	94	93	77	91
	Recall (%)	89	86	81	98	88	97	90
	F-measure (%)	94	87	89	96	90	86	91
DT	Precision(%)	75	65	67	87	77	70	75
	Recall (%)	79	67	68	85	78	65	75
	F-measure (%)	77	66	67	86	78	67	75
RF	Precision(%)	75	65	67	87	77	70	75
	Recall (%)	79	67	68	85	78	65	75
	F-measure (%)	77	66	67	86	78	67	75

TABLE 4. Comparative study between our method and the state of the art.

	Daylight	Nightlight
Psyllos et al. [35]	78.3	73.3
Petrovic et al. [14]	84.3	82.7
Peng et al. [17]	90.0	87.6
Dong et al. [36]	91.3	-
Peng et al. [37]	93.7	-
Dong et al. [18]	96.1	89.4
3D_VTC	97.3	93.1

information that represented by three-dimensional parameters allows the separation between ambiguous data for the five classifiers. The classification using only the gradient-based features gives the visual characteristics. Thus, they are not sufficient to distinguish between different types of vehicles. On the whole, SVM shows its superiority and gives better results because of its mechanism of classifying the data, and also for KNN and MLP given the obtained results.

In Table 4, The results of our 3D_VTC framework have been compared with the state of the art results. our framework shows its superiority compared with the reported methods in the literature.

V. CONCLUSION

In the present work, we propose an efficient framework for vehicle type classification 3D_VTC. This framework can accurately classify vehicles from different angles of view into six categories. In this work, we investigate the ability to enhance vehicle type classification results and to precisely describe the vehicles by combining the three-dimensional parameters delivered by the stereo vision system with the gradient-based features. A set of classifiers is used to learn vehicles types based on their feature vector to assign each vehicle to its corresponding type with a certain degree of confidence. The tests were performed on the BIT-dataset and our MoVITS-dataset, and demonstrate the effectiveness of our proposed framework. The global information is more significant and enhances the separability between classes. Also, the use of the gradient-based feature extraction was beneficial for our model in terms of eliminating the luminance effect. It is fair to mention that the calibration process has a direct impact on the correctness of the estimated 3D parameters. Hence, it needs to be done carefully.

REFERENCES

[1] O. Bourja, K. Kabbaj, H. Derrouz, A. El Bouziady, R. O. H. Thami, Y. Zennayi, and F. Bourzeix, "MoVITS: Moroccan video intelligent transport system," in *Proc. IEEE 5th Int. Congr. Inf. Sci. Technol.*, Oct. 2018, pp. 502–507. [Online]. Available: https://ieeexplore.ieee.org/document/8596566/

[2] A. El Bouziady, R. O. H. Thami, M. Ghogho, O. Bourja, and S. El Fkihi, "Vehicle speed estimation using extracted SURF features from stereo images," in *Proc. Int. Conf. Intell. Syst. Comput. Vis.*, Apr. 2018, pp. 1–6.

[3] J. Ren and W. Hanbo, "Application of stereo vision technology in 3D reconstruction of traffic objects," in *Proc. 17th Int. Symp. Distrib. Comput. Appl. Bus. Eng. Sci.*, Oct. 2018, pp. 100–102.

[4] Y. Cai, X. Chen, H. Wang, and L. Chen, "Deep representation and stereo vision based vehicle detection," in *Proc. IEEE Int. Conf. Cyber Technol. Automat., Control, Intell. Syst. (CYBER)*, Jun. 2015, pp. 305–310.

[5] Z. Dong, Y. Wu, M. Pei, and Y. Jia, "Vehicle type classification using a semisupervised convolutional neural network," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 2247–2256, Aug. 2015.

[6] H. Dong, X. Wang, C. Zhang, R. He, L. Jia, and Y. Qin, "Improved robust vehicle detection and identification based on single magnetic sensor," *IEEE Access*, vol. 6, pp. 5247–5255, 2018.

[7] C. Zhang, X. Chen, and W.-B. Chen, "A PCA-based vehicle classification framework," in *Proc. 22nd Int. Conf. Data Eng. Workshops*, Apr. 2006, p. 17. [Online]. Available: http://ieeexplore.ieee.org/document/1623812/

[8] P. Ji, L. Jin, and X. Li, "Vision-based vehicle type classification using partial Gabor filter bank," in *Proc. IEEE Int. Conf. Automat. Logistics*, Aug. 2007, pp. 1037–1040. [Online]. Available: http://ieeexplore.ieee.org/document/4338720/

[9] Y. Shan, H. S. Sawhney, and R. Kumar, "Unsupervised learning of discriminative edge measures for vehicle matching between nonoverlapping cameras," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 4, pp. 700–711, Apr. 2008. [Online]. Available: http://ieeexplore.ieee.org/document/4359343/

[10] M. Jiang and H. Li. (Apr. 2014). *Vehicle Classification Based on Hierarchical Support Vector Machine* [Online]. Available: http://link.springer.com/10.1007/978-3-319-01766-2_68

[11] A. H. S. Lai, G. S. K. Fung, and N. H. C. Yung, "Vehicle type classification from visual-based dimension estimation," in *Proc. IEEE Intell. Transp. Syst.*, Aug. 2001, pp. 201–206.

[12] S. Gupte, O. Masoud, R. F. K. Martin, and N. P. Papanikolopoulos, "Detection and classification of vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 1, pp. 37–47, Mar. 2002. [Online]. Available: http://ieeexplore.ieee.org/document/994794/

[13] J.-W. Hsieh, S.-H. Yu, Y.-S. Chen, and W.-F. Hu, "Automatic traffic surveillance system for vehicle tracking and classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 2, pp. 175–187, Jun. 2006. [Online]. Available: http://ieeexplore.ieee.org/document/1637673/

[14] V. Petrovic and T. F. Cootes, "Analysis of features for rigid structure vehicle type recognition," in *Proc. BMVC*, Sep. 2004, pp. 587–596. [Online]. Available: http://www.isbe.man.ac.uk/~bim/Papers/petrovic_bmvc04.pdf

[15] P. Negri, X. Clady, M. Milgram, and R. Poulenard, "An oriented-contour point based voting algorithm for vehicle type classification," in *Proc. 18th Int. Conf. Pattern Recognit.*, Aug. 2006, pp. 574–577. [Online]. Available: http://ieeexplore.ieee.org/document/1698958/

[16] B. Zhang, "Reliable classification of vehicle types based on cascade classifier ensembles," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 322–332, Mar. 2013. [Online]. Available: http://ieeexplore.ieee.org/document/6295662/

[17] Y. Peng, J. S. Jin, S. Luo, M. Xu, and Y. Cui, "Vehicle type classification using PCA with self-clustering," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops*, Jul. 2012, pp. 384–389.

[18] D. Marr and T. Poggio, "A computational theory of human stereo vision," *Proc. Roy. Soc. London. Ser. B, Biol. Sci.*, vol. 204, pp. 301–328, May 1979.

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.

[20] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, Nov. 2014, pp. 3320–3328.

[21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 886–893.

[22] I. Jolliffe, "Principal component analysis," in *Proc. Int. Encyclopedia Stat. Sci.*, Dec. 2011, pp. 1094–1096.

[23] O. Chapelle, P. Haffner, and V. N. Vapnik, "Support vector machines for histogram-based image classification," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 1055–1064, Sep. 1999.

[24] M.-L. Zhang and Z.-H. Zhou, "A κ-nearest neighbor based algorithm for multi-label classification," in *Proc. IEEE Int. Conf. Granular Comput.*, Jul. 2005, pp. 718–721.

[25] Y. Freund and L. Mason, "The alternating decision tree learning algorithm," in *Proc. ICML*, Jun. 1999, pp. 124–133.

[26] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[27] S. S. Haykin, *Neural Networks and Learning Machines*. London, U.K.: Pearson Upper Saddle River, 2009, vol. 3.

[28] E. Oyallon and J. Rabin, "An analysis of the SURF method," *Image Process. On Line*, vol. 5, pp. 176–218, Jul. 2015.

[29] R. I. Hartley and P. Sturm, "Triangulation," *Comput. Vis. Image Understand.*, vol. 68, no. 2, pp. 146–157, Nov. 1997.

[30] Z. Chen, N. Pears, M. Freeman, and J. Austin, "Road vehicle classification using support vector machines," in *Proc. IEEE Int. Conf. Intell. Comput. Intell. Syst.*, Nov. 2009, pp. 214–218.

[31] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," Oct. 2015, *arXiv:1510.00149*. [Online]. Available: https://arxiv.org/abs/1510.00149

[32] S. Targ, D. Almeida, and K. Lyman, "Resnet in Resnet: Generalizing residual architectures," Mar. 2016, *arXiv:1603.08029*. [Online]. Available: https://arxiv.org/abs/1603.08029

[33] FHWA. (2014). *Introduction to Vehicle Classification*. [Online]. Available: https://www.fhwa.dot.gov

[34] A. Psyllos, C. N. Anagnostopoulos, and E. Kayafas, "Vehicle model recognition from frontal view image measurements," *Comput. Standards Interfaces*, vol. 33, no. 2, pp. 142–151, Feb. 2011. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0920548910000838

[35] Z. Dong and Y. Jia, "Vehicle type classification using distributions of structural and appearance-based features," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2013, pp. 4321–4324.

[36] Y. Peng, J. S. Jin, S. Luo, M. Xu, S. Au, Z. Zhang, and Y. Cui, "Vehicle type classification using data mining techniques," in *Era Interact. Media*. New York, NY, USA: Springer, 2013, pp. 325–335.

HATIM DERROUZ (M'18) was born in Ouezzane, Morocco, in 1992. He received the License in computer science from the University of Ibn Tofail, Kenitra, Morocco, in 2014, and the M.Sc. degree in image processing and business intelligence from Sidi Mohamed Ben Abdellah University, Fes, Morocco, in 2016. He is currently pursuing the Ph.D. degree with the Higher National School of Computer Science and System Analysis (ENSIAS), University of Mohamed V, Rabat, Morocco.

From 2017 to 2019, he was a Research Student with the Moroccan Foundation for Advanced Science, Innovation & Research. His research interests include machine learning, deep learning, pattern recognition, and computer vision applied to the intelligent transport system and other applications.

ABDERRAHIM ELBOUZIADY was a Computer Science Engineer with the High School of Applied Science at Marrakesh, in 2016. He is currently a Ph.D. Researcher with the Embedded System Team, MAScIR, and the Advanced Digital Enterprise Modeling and Information Retrieval Laboratory, Information Retrieval and Data Analytics (IRDA) Team, ENSIAS. His Ph.D. thesis is on development of image processing algorithms for vehicle speed estimation and traffic violation detection (speed exceeding, red light running, stop running, and so on).

HAMD AIT ABDELALI received the master's degree in computer science and telecommunications from the Faculty of Sciences, Mohammed V University, Rabat, Morocco, in 2012. He defended his thesis at the Faculty of Rabat, in 2016. Since 2016, he has been an Associate Member of the Laboratory for Research in Computing and Telecommunications (LRIT), Faculty of Sciences of Rabat, Morocco. He is currently holding a postdoctoral position in Intelligent Transport Systems (ITS) at the Embedded Systems Team, MAScIR. His research interests include artificial intelligence, deep learning, computer vision, embedded systems, algorithms, behavior analysis, event detection, traffic offense detection, video/image processing, and pattern recognition.

RACHID OULAD HAJ THAMI received the Ph.D. degree in computer science from the Faculty of Sciences Ben M'Sik Sidi Otthman, Casablanca, Morocco, in 2002. He is currently a Full Professor of computer engineering with ENSIAS, Rabat IT Center, Mohammed V University, Rabat, Morocco. His research interests include multimedia and information retrieval, image and video analysis, intelligent video surveillance, and health applications.

SANAA EL FKIHI received the Ph.D. degree in computer science from Mohammed V Rabat University, Rabat, Morocco, and the University of Science and Technology of Lille, in 2008. She is currently a Full Professor of Computer Engineering with ENSIAS, Rabat IT Center, Mohammed V University. Her research interests include multimedia and information retrieval, image and video analysis, intelligent video surveillance, and health applications.

FRANÇOIS BOURZEIX received the DEA degree in applied mathematics from University PARIS IX Dauphine, in 1995, and the Engineering degree in electronics and signal processing from the National School of Telecommunications of Paris, in 1996. He was an Engineer with the Polytechnic School of Paris, in 1994. He has worked in several high tech companies, including Motorola Semiconductor as a Software Development Engineer and a Hardware and Architectural Engineer of wireless products (2.5G, 3G, and Wi-Fi), ST Microelectronics as the Technical Leader in firmware development for HDMI, and Amesys Morocco applications as the Technical Director. In 2009, he joined MAScIR. He is currently the Director of the Embedded Systems Department, Moroccan Foundation for Advanced Science, Innovation & Research.