Automatic Road Crack Detection and Characterization

Henrique Oliveira, *Member, IEEE*, and Paulo Lobato Correia, *Senior Member, IEEE*

***Abstract*—A fully integrated system for the automatic detection and characterization of cracks in road flexible pavement surfaces, which does not require manually labeled samples, is proposed to minimize the human subjectivity resulting from traditional visual surveys. The first task addressed, i.e., crack detection, is based on a learning from samples paradigm, where a subset of the available image database is automatically selected and used for unsupervised training of the system. The system classifies nonoverlapping image blocks as either containing crack pixels or not. The second task deals with crack type characterization, for which another classification system is constructed, to characterize the detected cracks' connect components. Cracks are labeled according to the types defined in the Portuguese Distress Catalog, with each different crack present in a given image receiving the appropriate label. Moreover, a novel methodology for the assignment of crack severity levels is introduced, computing an estimate for the width of each detected crack. Experimental crack detection and characterization results are presented based on images captured during a visual road pavement surface survey over Portuguese roads, with promising results. This is shown by the quantitative evaluation methodology introduced for the evaluation of this type of system, including a comparison with human experts' manual labeling results.**

***Index Terms*—Clustering, crack characterization, crack severity level, road crack detection, segmentation, unsupervised learning.**

I. Introduction

CRACKS are the most common road pavement surface defects, and capturing imagery of pavement surface during road surveys, for crack detection and characterization, is considered an adequate procedure for the collection of data about the condition of the pavement surface [1], [2]. Automatic crack detection and characterization systems are being developed for fast and reliable pavement surface defect analysis, instead of relying solely in the slower and subjective traditional human inspection procedures, also contributing to the development of a safer survey methodology, particularly when monitoring high-speed roads like highways [3]. These procedures can be combined with the usage of intelligent transportation systems, which can detect and identify vehicle types (e.g., heavy vehicles like tractor trailers) [4], together with the monitoring of traffic intensity [5], for instance, to help identify roads where more frequent pavement status surveys are required.

Automatic procedures relying on neural networks, Markov random fields, edge detectors, or morphological operators have been used to characterize road pavement surface distresses like cracking [6], as discussed in Section II.

The fully integrated system for crack detection and characterization proposed here, i.e., crack detection tool of Instituto de Telecomunicações (IT) (CrackIT), requires no manually provided labeled examples for the training phase, automatically selecting training images, and learning the statistics about crack and noncrack image blocks. Only basic prior radiometric knowledge about crack characteristics is necessary (image regions darker than their surroundings). In fact, image samples for training the system are automatically selected, and the crack blocks they contain are preliminary labeled using a simple system that may miss some portions of the existing cracks but still is able to collect enough statistical information for the detection stage.

An unsupervised two-step pattern recognition system is presented. First, the detection of image blocks containing crack pixels is addressed. Then, cracks are characterized into types as specified by the Portuguese Distress Catalog [7], and CrackIT is able to classify the different crack types present in a given image, as well as to estimate the average width of each detected crack for a crack severity level assignment. The 2-D feature space used for crack detection considers two simple but effective local statistics: mean and standard deviation of gray level values inside nonoverlapping image blocks; then, six clustering techniques are confronted to infer about their crack detection suitability. Crack type characterization exploits the spatial distribution of image blocks composing the detected cracks' connected components using another 2-D feature space. Finally, a novel crack severity assignment procedure is applied, automatically calculating the average width of each crack.

The proposed crack detection strategy follows a learning from samples paradigm, with unlabeled data being used for training the CrackIT system. The crack type and severity characterization steps require no training. Performance evaluation is conducted using pavement surface images captured during a visual survey along a Portuguese road, a subset of which has been manually labeled by human experts, which will be taken as ground truth, for comparison purposes.

This paper is structured as follows. After this brief introduction to the crack detection and characterization problem, Section II presents an overview of the relevant literature.

Manuscript received October 6, 2011; revised March 1, 2012 and May 2, 2012; accepted July 2, 2012. Date of publication August 7, 2012; date of current version February 25, 2013. This work was supported by Fundação para a Ciência e Tecnologia and Instituto de Telecomunicações under Project PEst-OE/EEI/LA0008/2011. The Associate Editor for this paper was Prof. Q. Ji.

H. Oliveira is with Instituto de Telecomunicações–Instituto Superior Técnico, Lisbon 1049-001, Portugal, and also with Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Beja, Beja 7800-295, Portugal (e-mail: hjmo@lx.it.pt).

P. L. Correia is with Instituto de Telecomunicações–Instituto Superior Técnico, Lisbon 1049-001, Portugal (e-mail: plc@lx.it.pt).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2012.2208630

Section III describes the proposed crack detection approach, notably in terms of parameter estimation and the classifiers used. Section IV presents the proposed crack type characterization and severity level assignment strategies. Experimental results are discussed in Section V, and Section VI draws conclusions and provides hints for future work.

II. Literature Review

The number of recently published papers dealing with crack detection and characterization of pavement surface distresses shows an increasing interest in this area. A recent publication [8] proposes a methodology to detect cracks using a multiscale approach based on Markov random fields to segment fine structures (cracks) in road pavement surface images. Cracks are enhanced using a 1-D Gaussian smoothing filter and then processed by a 2-D matched filter to detect them. A total of 64 road pavement surface images representing several crack types are considered for experimentation, producing a qualitative evaluation. Details on image characteristics or the type of sensor used to capture them are not provided.

In [9], crack detection starts with a nonlinear filter to enhance the contrast between distresses and the image background. Then, segmentation by thresholding and finally a pixel-based connectivity analysis is performed to identify cracks. The actual crack detection performance is difficult to evaluate as no objective evaluation metrics are used, and a global analysis of the image database is not included.

A fractional differential and wavelet transform is presented in [10], being qualitatively compared with other edge detection operators like Sobel, Prewitt, and Logarithm of Gaussian (LoG), to show its superior performance. A Sobel edge detector after image smoothing with a bidimensional empirical mode decomposition is proposed in [11] to detect cracks. The reported results use 15 images, but no details about image spatial resolution are provided, and the results make evident some of the challenges faced by these types of systems, with false crack region edges appearing due to the presence of white lane markings, and irregularities (more difficult to handle) always present in this type of pavement surface due to the appearance of coarse surface aggregates caused by the loss of fine aggregates (raveling). Another author [12] evidences the difficulty of detecting cracks of less than 3 mm width when using edge detectors. A nonsubsampled contourlet transform is adopted in [13] to detect cracks, wherein a limited set of experimental results is presented.

Other approaches, based on segmentation by thresholding to automatically detect cracks, are proposed in [14] and [15], where results using the Otsu threshold selection method [16] and the entropic method of Kapur [17] are compared with the neighboring difference histogram method proposed in [14]. The authors of [15] claim that their segment extending approach achieves better results because they are capable of dealing with noisy images and detecting unclear and disconnected cracks, but no comparative results are included, and information about the ground truth used is omitted. Another approach extracts linear features (cracks) [18] comparing two methodologies: one based on holistic thresholding and the second employing the Otsu algorithm. The second methodology proves to be more efficient in terms of both the threshold value and the required computational effort for crack detection. Again, no details about the image database are provided, and the analysis of results is qualitative and based on a single image. An improved dynamic programming (DP)-based algorithm is proposed in [19] to automatically segment pavement distresses, where the authors claim that their algorithm performs three times faster than the original DP-based method while providing the same accuracy. No severity levels are assigned for the segmented cracks.

A complete methodology to automatically detect and characterize pavement defects is proposed in [20], using grayscale images captured by line scan cameras illuminated by lasers during road surveys performed using a high-speed image acquisition system. Crack detection uses a conditional texture anisotropy measurement to each image individually, and defect characterization uses a multilayer perceptron neural network with two hidden layers. The results presented are promising, but the experimental evaluation does not support the distinction of multiple cracks in the same image.

Another neural-network-based crack imaging system to classify cracks is proposed in [21], where a proximity-based neural network, trained with a computer-generated artificial data set, following the Federal Highway Administration guidelines, comprising 60 images for each crack type (longitudinal, transversal, blocks, alligator, and no crack), performed well in comparison with image- and histogram-based neural networks. Although the reported results are good using 124 real images (for which no details or acquisition information is provided), there is no indication if the system is capable of dealing with more than one distinct crack type per image, and no severity levels are assigned to the detected cracks.

In summary, most of the automatic crack detection strategies reported in the literature, as indicated in [22], where a classification of existing methods is also proposed, perform a pixel-based analysis of road pavement surface images, producing global image analysis results, to indicate the type of distress detected in that image. Usually, just a qualitative analysis of the results is done.

This paper proposes to overcome some of the identified limitations, performing a (fast) block-based crack detection capable of identifying the presence of multiple cracks in a given image and providing a detailed quantitative evaluation of the results obtained, highlighting which blocks in each image contain cracks, identifying the type of each detected crack, and assigning it a severity level. The obtained results are quantitatively evaluated using a set of well-known metrics and taking into account a ground truth provided by a skilled operator who manually labeled a subset of the database images, identifying all blocks in each of those images as containing crack pixels or not.

III. Unsupervised Crack Detection

The proposed automatic system (CrackIT) to detect and characterize cracks found in flexible road pavement surfaces follows a block-based image analysis approach, and its general architecture is illustrated in Fig. 1.

The image database considered is composed by gray-level images of fixed size (1536×2048 pixels) with pixel intensities

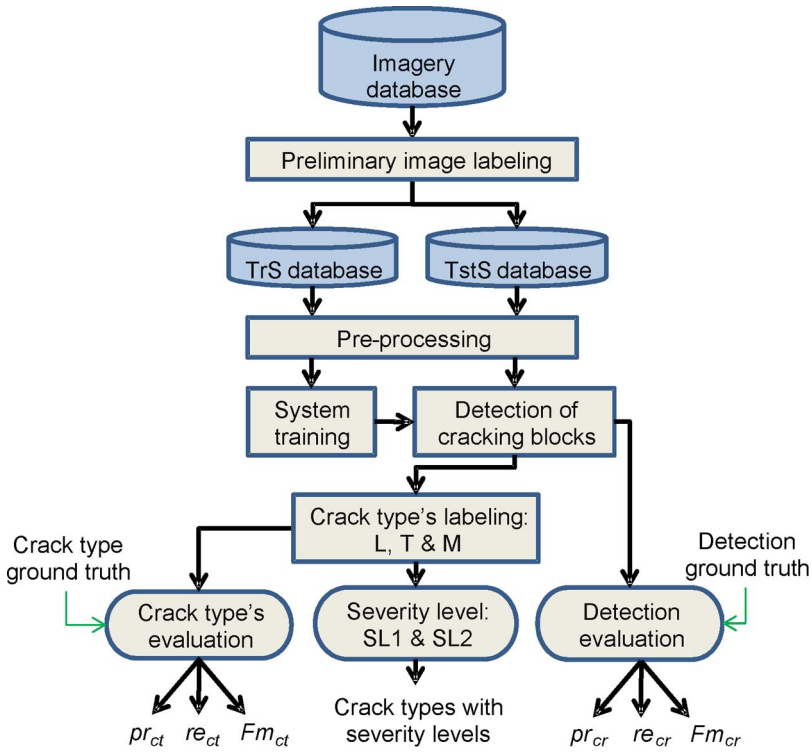

Fig. 1. Proposed CrackIT system's architecture.

ranging from 0 (black) to 255 (white). They were captured by a digital camera at sunlight with its optical axis perpendicular to the road surface and its lateral edges parallel to the road axis. Each pixel corresponds to a road area of about 1 mm^2. For processing purposes, nonoverlapping 75 $\times$ 75 pixel blocks were considered, as this represents a good compromise between computational performance (faster processing time and lower memory storage requirements) and accuracy of the detection and crack characterization results [23]—notice that blocks that are too small tend to increase the number of false positive crack detections, while for blocks that are too large, the impact of small cracks tends to vanish in the computation of statistical features. Two 20 $\times$ 27 feature matrices are computed for each image: 1) the mean matrix (M_m), with each block's average intensity, and 2) the standard deviation matrix (STD_m), with the corresponding standard deviation (std) values. The block-based analysis ignores the incomplete blocks at the right (corresponding to part of the curb) and at the bottom (which is repeated at the top of the following image) sides of the images.

In the crack detection task, a training stage is implemented, where automatically selected image samples are used for training the system, composing the training set (TrS), whereas the remaining database images are used for testing, composing the test set $(TstS)$, to provide a quantitative measure of the crack detection quality. This strategy envisages the estimation of more reliable parameters of the probability density functions used to model the stochastic pattern recognition system used for crack detection, allowing the system to adjust itself to the different textural characteristics found in the surfaces of different roads.

The rest of this section details the main steps of the crack detection stage according to the system architecture in Fig. 1. Crack type characterization and assignment of severity levels are discussed in Section IV.

A. Preliminary Image Labeling

The proposed algorithm starts by automatically splitting the image database into two subsets: TrS and $Tsts$. The selection of TrS images, i.e., those images to be used for training the system, is based on the analysis of the image's mean matrices (M_m). The methodology used, presented in [23], considers that training images should contain relevant road pavement cracks, i.e., images allowing an easy detection of a sufficient number of "crack" blocks during the preliminary labeling stage, to ensure an adequate learning stage. M_m matrices are vertically and horizontally scanned to find blocks with relevant crack pixels, producing binary matrices (one per image), where blocks containing evident crack pixels are labeled "1," and the remaining are labeled "0." A sample of the results obtained after this preliminary image labeling procedure is included in Fig. 2.

Equations (1) and (2) are used for the preliminary labeling of blocks (plb) along the horizontal and vertical scannings of the rows and columns of the M_m matrix, respectively, i.e.,

$$plb_H^{(i,j)} = \left[std\left(Ah^{(i,j)}\right) > \left(k_1 \times std(Bh^i) + k_2 \times \text{mean}(Bh^i)\right)\right] \wedge \left[\left(Ah^{(i,j)}[1] - Ah^{(i,j)}[2]\right) > 0\right] \quad (1)$$

$$plb_V^{(i,j)} = \left[std\left(Av^{(i,j)}\right) > \left(k_1 \times std(Bv^j) + k_2 \times \text{mean}(Bv^j)\right)\right] \wedge \left[\left(Av^{(i,j)}[1] - Av^{(i,j)}[2]\right) > 0\right] \quad (2)$$

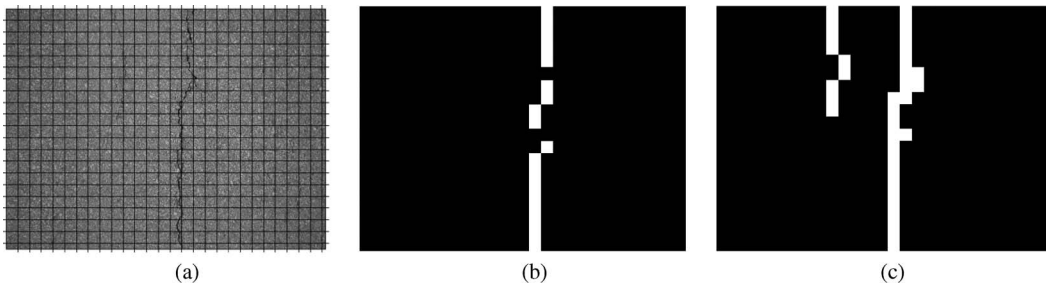

Fig. 2. (a) Original image, divided into nonoverlapping blocks of 75 × 75 pixels, (b) preliminary labeling of blocks with relevant crack pixels, and (c) the corresponding ground truth.

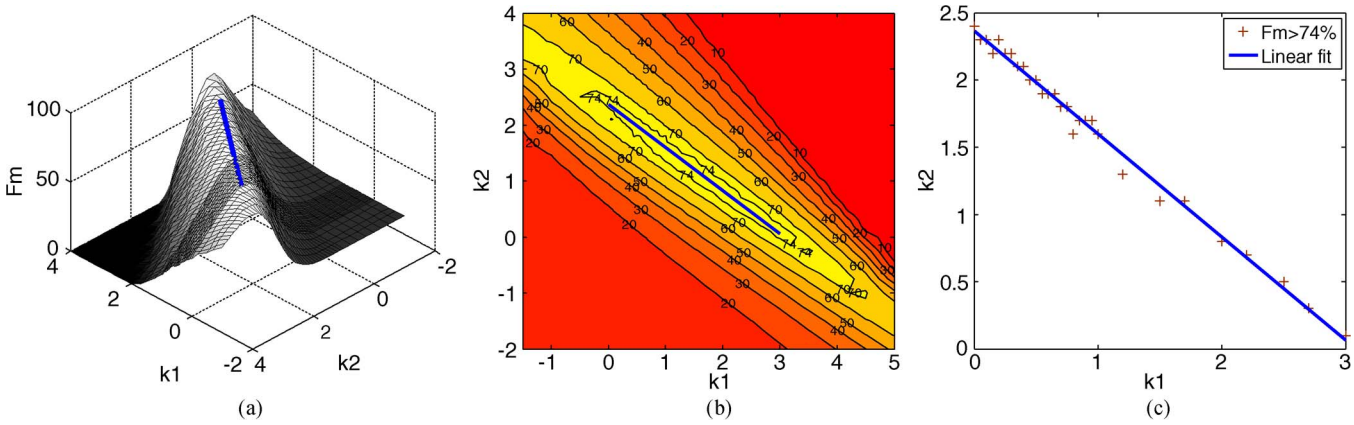

Fig. 3. (a) Three-dimensional representation of Fm values as a function of k_1 and k_2 parameters and (b) the corresponding isolines. (c) Linear fitting for k_1 and k_2, with red points corresponding to simulations leading to a Fm value higher than 74.0%.

where i and j are the row and column indexes, and A_h, B_h, A_v, and B_v are given by

$$Ah^{(i,j)} = \left[\frac{M_{m(i,j-1)} + M_{m(i,j+1)}}{2};\ M_{m(i,j)}\right]$$
$$Bh^{i} = \left[0;\ std\left(Ah^{(i,2)}\right);\ \ldots\ std\left(Ah^{(i,26)}\right);\ 0\right] \quad (3)$$

$$Av^{(i,j)} = \begin{bmatrix}\frac{M_{m(i-1,j)} + M_{m(i+1,j)}}{2} \\ M_{m(i,j)}\end{bmatrix},\ Bv^{j} = \begin{bmatrix}0 \\ std\left(Av^{(2,j)}\right) \\ \vdots \\ std\left(Av^{(19,j)}\right) \\ 0\end{bmatrix} \quad (4)$$

with the values of parameters k_1 and k_2 for optimal operation being related to each other. This relationship can be determined from the TrS set by analysis of the F-measure (Fm) metric, which is defined as

$$Fm = \frac{2 \times precision \times recall}{precision + recall} \quad (5)$$

where $precision$ is the rate between the number of crack blocks correctly detected and the total amount of blocks detected as cracks, and $recall$ stands for the rate between the number of crack blocks correctly detected and the number of crack blocks in the ground truth [24].

Intuitively, the "A" matrices register each block's mean intensity as well as the average of the block's immediate neighbor mean intensities. The local mean intensity value variation can thus be computed by $std(A)$. Those local variations (of blocks) can then be compared against the average variation and the standard deviation observed along a line/column. All local variation values along a line/column are stored in the "B" matrices. This is the basis for identifying the prominent crack blocks in the preliminary labeling stage.

Fig. 3 shows the Fm results obtained after 2009 simulations of the preliminary labeling using all database images for which a ground truth has been manually provided by a skilled technician, who labeled all image blocks containing crack pixels. The best Fm values for the preliminary labeling step (higher than 74%) are obtained for k_1 and k_2 values in the ranges [0;3] and [0;2.5], respectively [see the isoline labeled 74 in Fig. 3(b)], leading to a linear fit [see line in Fig. 3(c)] represented by

$$k_1 = -0.7671 \times k_2 + 2.3636. \quad (6)$$

For instance, the results presented in Fig. 2(b) were obtained using 0.4 and 2.1 for k_1 and k_2, respectively. Notice that the k_1 and k_2 values suggested by the CrackIT application (and which may be manually adjusted by the system operator) always propose to use k_1 smaller than k_2 to favor *recall* instead of *precision*, as for this type of application the more important is not to miss any existing road crack (see Section V). Moreover, for different types of pavement surfaces, different values for the k_1 and k_2 parameters may be selected.

Both plb_H and plb_V (binary matrices storing the partial preliminary labeling results) are merged after removing isolated blocks in each of them separately, i.e.,

$$plb^{(i,j)} = plb_H^{(i,j)} \vee plb_V^{(i,j)}. \quad (7)$$

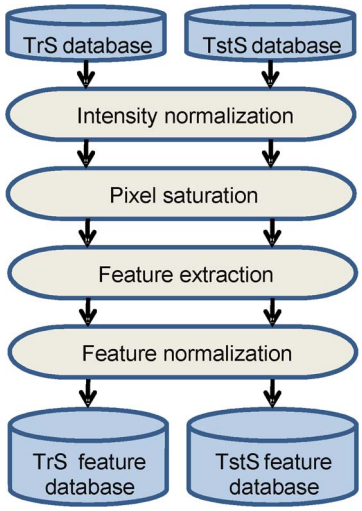

Fig. 4. Proposed CrackIT preprocessing step.

A preliminary labeling binary matrix (plb) is thus obtained for each image, and the length of each image's longest connect component (lcc), after applying a connected components algorithm, is computed. The preliminary labeling step ends after sorting all the images in descending order of their lcc value.

Now, the training set TrS can be automatically selected, being composed by a number of images (controlled by the system operator) taken from the top of this sorted list. Once the composition of TrS and $TstS$ sets is known, both are further preprocessed to keep the adequate level of detail for crack detection and characterization, as described next.

B. Preprocessing

The proposed preprocessing step follows the architecture illustrated in Fig. 4. This is an extended version of the strategy adopted in [23], envisaging better crack detection and characterization results.

Intensity Normalization and Pixel Saturation: Preprocessing starts with pixel intensity normalization, independently applied to each image, to deal with the nonuniform background illumination, which is mainly due to the type of sensor used for image capture. The goal is to obtain the same average pixel intensity for blocks preliminary labeled with "0" (considered as background) in each image, whereas the remaining (crack) blocks keep a lower average intensity due to the presence of relevant (darker) crack pixels. The preliminary labeling step needs to be done prior to this normalization as an initial location of crack blocks is crucial to prevent a substantial increase of pixel intensity values in blocks preliminary classified as containing cracks and thus to enhance the subsequent detection task performance [23].

Road pavement surface images often include bright pixels due to specular reflections on some surface materials. That increases the standard deviation of pixel intensities in blocks without cracks, eventually leading to similar std values in both crack and noncrack blocks, which is undesirable since this paper intends to explore std as a feature for crack detection.

A simple way to handle this problem consists of applying a saturation function, such that pixels with intensities higher than a certain threshold (automatically selected as the average of all pixel intensities in each image) are replaced by the threshold value. This results in slightly darker images, without losing information regarding the presence of cracks, as notably, the "std" feature becomes more discriminative [23].

Feature Extraction and Normalization: The proposed crack detection step relies on two simple but effective features: the mean and the standard deviation values of pixel-normalized intensities within an image block. The resulting 2-D feature space is illustrated in Fig. 5(a) for the input image of Fig. 2(a), where each point identifies one image block. To reduce the scattering of the computed features among images, due to different individual average intensities, which would influence negatively the classification results, also a feature space normalization procedure is proposed. Each database image undergoes three actions: 1) a clustering analysis to distinguish between target (no crack) and outlier (crack) points, 2) computation of a global centroid for the target cluster, and 3) linear fitting applied to the target points to compute a reference angle for each 2-D feature space (being the lowest angle between the line and the horizontal axis).

All the measurements (mean and standard deviation of gray level values within a block) for each image compose a pattern vector $\mathbf{x}$, representing a sample of the random variable $\mathbf{X}$, taking values on a sample space $\boldsymbol{X}$. For each element x_i of the pattern vector $\mathbf{x}$, one possible class y_i is assigned, where $\mathbf{Y}$ is the class set. Thus, the feature set is

$$F = \{(x_1, y_1) \ldots (x_n, y_n) : x_i \in \Re^2; y_j \in \{c_1, c_2\}\} \quad (8)$$

where n is the number of points for the pattern vector $\mathbf{x}$, and j is the number of classes, with x_i being the feature values extracted from an image belonging to the TrS set. Only two classes are considered: 1) blocks without cracks, labeled as class c_1; and 2) blocks with cracks, labeled as class c_2.

The clustering analysis adopts a one-class classification approach, and according to the terminology in use, the target class c_1 (no crack) is assumed to be well defined in opposition to class c_2 (crack), called the outlier class, whose samples are sparse. Assuming a Gaussian distribution for the target class, the decision rule is based on the Mahalanobis distance, instead of the density estimate, to avoid numerical instabilities [24], i.e.,

$$\begin{cases} x_i \text{ is outlier if } (x_i - \mu)^{\mathbf{T}} \sum^{-1} (x_i - \mu) > \varepsilon \\ \text{otherwise, } x_i \text{ is a detected target} \end{cases} \quad (9)$$

with μ (the coordinates of center of mass in the 2-D feature space) and Σ being, respectively, the mean vector and the general covariance matrix, whereas ε represents the decision threshold. The clustering results for the TrS set will be used for training the CrackIT system. A sample clustering result, obtained for the input image of Fig. 2(a), is shown in Fig. 5(b) superimposed with the centroid being computed only using target points (points belonging to class c_1) using a decision threshold of 0.05 (observed to be the global error rate when evaluating the preliminary labeling of image regions, see Section V).

Next, the target cluster global centroid is computed, and all the images' feature spaces are aligned by translation of their

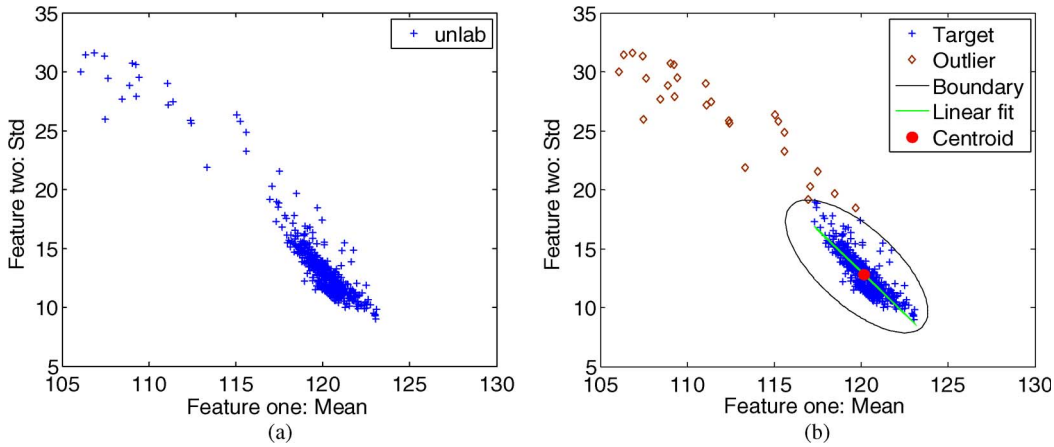

Fig. 5. (a) Two-dimensional feature space for the image in Fig. 2(a), with unlabeled points. (b) Target and outliers after clustering, also showing linear fitting results.

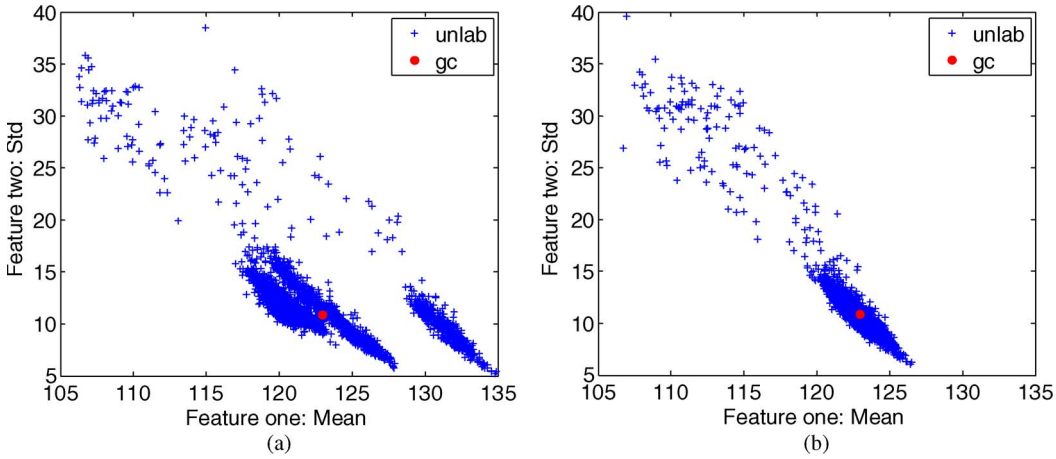

Fig. 6. Two-dimensional feature space for TrS (a) before and (b) after feature normalization, with the global centroid (gc).

respective centroids toward the global centroid. Finally, the 2-D feature spaces are rotated to align them with a global reference angle. Feature space normalization sample results, using a TrS composed of five images, are included in Fig. 6.

Analyzing the right plot of Fig. 6, a well-defined cluster of points can be identified on its bottom right, i.e., blocks that may be classified as not containing crack pixels, while maintaining the scattering of points related to blocks with crack pixels. Therefore, the normalized feature space allows a better distinction between the considered classes, whereas before feature normalization, more than one cluster of target blocks could be identified. Applying (5) to this example, Fm values of 89% and 97% were achieved for the nonnormalized and normalized feature spaces, respectively.

C. System Training

For crack detection purposes, a *learning from examples* paradigm is implemented, and a boundary decision is estimated based on training samples, i.e., the features extracted from images composing the TrS set. Six unsupervised classification strategies are tested and evaluated to ascertain their suitability for the crack detection task. Three of these strategies are based on clustering techniques, whereas the remaining are one-class classification strategies. In all cases, unlabeled data are used for training the detection task, with no ground truth data being used at this stage.

The clustering techniques aim to group image blocks with similar characteristics. Notably, blocks without crack pixels are expected to show smaller feature value variations, whereas blocks containing cracks are expected to present higher pixel intensity std values, as well as lower mean intensity values.

Training Using Clustering Techniques: Three clustering techniques were considered [25]: 1) hierarchical; 2) *k-means* (based on Euclidean distances) using the *KCentres* algorithm as initialization step; and 3) the mixture of two Gaussians [Gaussian mixture model (GMM)]. Each of them tries to identify two clusters on the data, corresponding to classes c_1 and c_2.

After labeling each point in the feature space as belonging to one of the clusters, a quadratic boundary decision is computed using the assigned labels. For this purpose, a bidimensional Gaussian probability density function is assumed to be a good model for the feature space data, as shown by the results included in the following. Moreover, the assumption of a class-condition normal distribution leads to a better performance of the system, measured by Fm, during the training phase, as shown next.

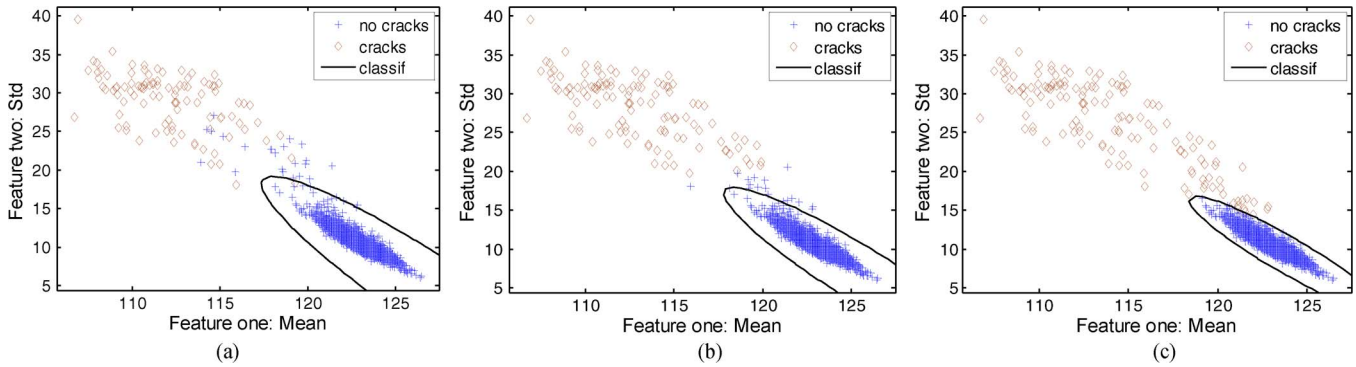

Fig. 7. Clustering results computed using the 2-D feature space extracted from TrS (point labels assigned after each clustering technique) and the computed quadratic Gaussian decision boundary. (a) Hierarchical clustering. (b) k-means clustering. (c) Mixture of two Gaussians clustering.

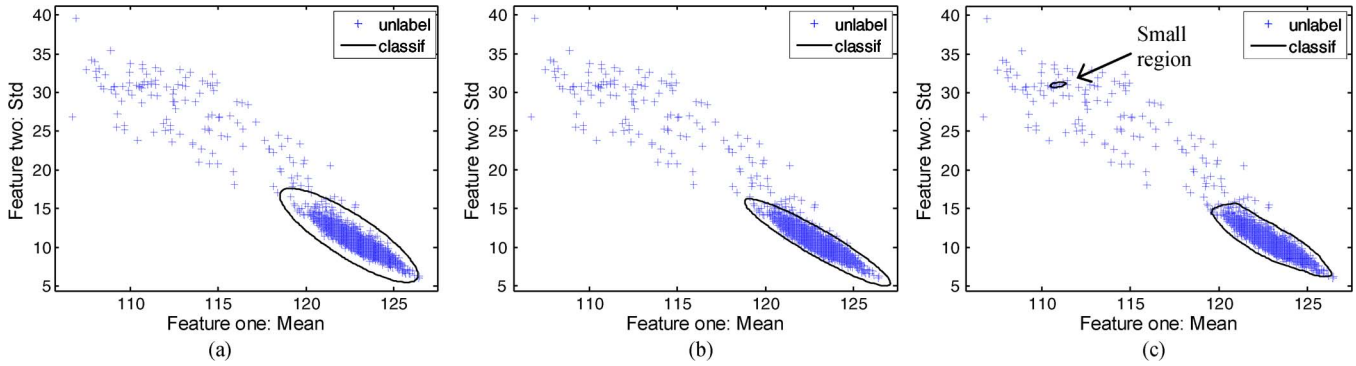

Fig. 8. (a) One-class classification results using Gaussian density estimation of the unlabeled points, computed for the TrS, with $\varepsilon = 0.05$. Points inside the boundary decision are considered as target points (a well-defined cluster), whereas the remaining points are the outliers. (b) One-class MCDG estimate results using a Gaussian model of the unlabeled points, computed for the TrS, and the computed boundary decision. (c) One-class Parzen density estimator results of the unlabeled points, computed for the TrS, and the computed decision boundaries.

Fig. 7 shows the class boundaries computed using the labeled points resulting from applying each of the mentioned clustering techniques to the TrS 2-D feature space. All points were then relabeled according to the computed boundary. This resulted in increased Fm values: notably from 87% (with the initial cluster based labeling) to 97.5% (decision boundary based labeling) for the hierarchical clustering case, and from 94% (cluster based) to 97.5% (decision boundary based) for k—*means* clustering [see Fig. 7(a) and (b)]. For GMM clustering, an Fm value of 96.6% is obtained. In this case, the estimation-maximization algorithm exploits the use of well-known probability density functions, notably the Gaussian distributions, and the labels presented in Fig. 7(c) are obtained when the decision boundary (computed using the clustering technique) is applied to the unlabeled training data. The iterative process stops when the maximum likelihood estimation of the parameters of Gaussian distributions does not change significantly (i.e., a variation of less than 10%), or a maximum of 50 iterations is reached.

Training Using One-Class Classification Strategies: All the clustering approaches discussed so far assume, as prior knowledge, the existence of two clusters. In Fig. 7, one well-defined cluster of points can be identified (shown in blue), whereas the remaining points present a more scattered structure. The scattered points can be viewed as outliers of the well-defined cluster, which motivates the usage of one-class classification strategies, with only the main cluster being considered during the learning stage.

Density estimation, boundary, and reconstruction methods are the three main approaches to deal with the one-class classification problem [26]. The first one is used here, adopting a normal density for the well-defined target class (class c_1). The aim is to set a threshold, applied to the density estimation of the TrS data, to detect outliers (class c_2). The approach based on density estimation of the target class works well when the sample size is large enough and a flexible distribution model is used (like Gaussian and the Parzen estimation models), which is the case here [26].

Three one-class density methods are tested [24]: 1) a simple Gaussian density method; 2) a minimum covariance determinant Gaussian (MCDG) classifier; and 3) the Parzen density estimator. An Fm value of 96.7% was obtained when applying the Gaussian density method to the feature space extracted from TrS [see Fig. 8(a)], 92.3% for the case of MCDG [see Fig. 8(b)], and 92% when using the Parzen density estimator [see Fig. 8(c)].

D. Crack Detection

After estimating the decision boundary, using the features extracted from TrS images, the test stage begins. For each $TstS$ image, the estimated decision boundaries are superimposed to the computed 2-D feature points to identify the image blocks without crack pixels (class c_1) or with crack pixels (class c_2).

Fig. 9. Crack detection. (Left) Ground truth. (Middle) Preliminary labeling result. (Right) Crack detection result using *k-means* clustering followed by the elimination of isolated crack blocks.

Detection results are stored using binary matrices with "1s" and "0s" representing, respectively, blocks with and without crack pixels.

An additional step can be considered as an option to the system operator, consisting of the removal of isolated "crack" blocks. The main motivation here is that oil spots are often found in pavement surfaces, appearing as isolated clusters of dark pixels that could be erroneously detected as crack pixels, resulting in false-positive detections.

As an example, sample detection results for the k-means clustering (followed by the elimination of isolated crack blocks) are shown in Fig. 9 for the road surface image in Fig. 2(a). Moreover, ground truth and preliminary crack characterization results are included in Fig. 9.

IV. Crack Type Labeling and Severity Levels

In addition to detecting blocks containing crack pixels, the proposed system also aims to characterize them according to a subset of the cracking distress types defined in the Portuguese Distress Catalog [7] (see also Fig. 1): longitudinal (L), transversal (T), or miscellaneous (M). Moreover, each crack can be assigned a severity level mainly related to the width of the crack. The remainder of this section details the crack type classification and presents a novel severity level assignment procedure.

A. Crack Type Labeling

Crack type labeling begins with the application of a connected component algorithm to the crack detection results. Another 2-D feature space is used for this purpose using the standard deviations of the column (feature one) and row (feature two) coordinates of each crack connected component, as described in [27]. Fig. 10 illustrates the 2-D feature space used to label crack types, where d_x and d_y are, respectively, the distances to the horizontal and vertical axes, d_b is the distance to the feature space bissectrix (solid line), whereas the dash-dot lines represent the geometrical places wherein $d_x = d_b$ and $d_y = d_b$.

The crack type characterization rules are

$$\begin{aligned} &1)\ \text{longitudinal crack (class } c_L),\ \text{if } d_{y_i} < d_{b_i};\\ &2)\ \text{transversal crack (class } c_T),\ \text{if } d_{x_i} < d_{b_i};\\ &3)\ \text{miscellaneous crack (class } c_M),\ \text{if } d_{b_i} \leq d_{x_i} \text{ or } d_{b_i} \leq d_{y_i} \end{aligned} \tag{10}$$

where i is the connected component index (between 1 and the number of connected components detected in an image).

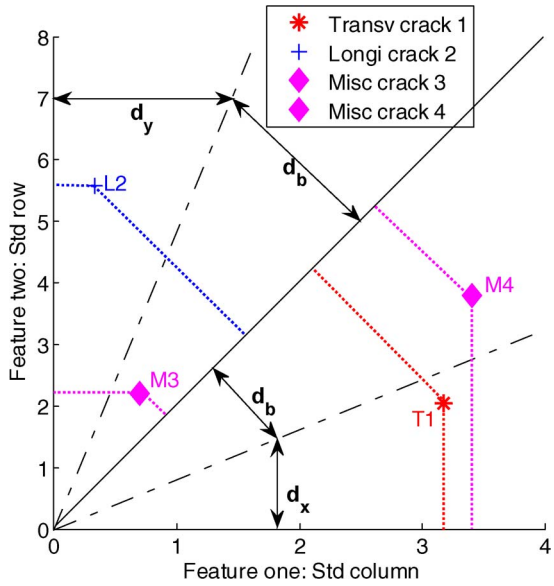

Fig. 10. Two-dimensional feature space used for crack type characterization.

B. Severity Level Assignment

The Portuguese Distress Catalog defines three levels of severity for cracks mainly related to their width [7]: level 1, for closed cracks with less than 2 mm width; level 2, for open cracks with more than 2 mm width, without presenting degraded edges; and level 3, for open cracks that also have more than 2 mm width but present degraded edges.

The present implementation uses the width of a detected crack to assign severity levels (see also Fig. 1): Level 1 (SL_1) is assigned to cracks with less than or equal to 2 mm width, and level 2 (SL_2) or level 3 (SL_3) are assigned to cracks of more than 2 mm width. Thus, the proposed methodology does not (currently) distinguish between crack severity levels 2 and 3.

The width of a crack is difficult to quantify, as it can be measured in various locations (e.g., top, half way, bottom) and it may not be uniform all along [28]. Here, the proposal is to calculate the average width of a crack segment (Wcs), working at pixel level, instead of the block level analysis adopted for the detection phase. Fig. 11 details the proposed crack severity level assignment procedure.

The first step is to segment the (darker) crack pixels by thresholding. In this proposal, two different thresholds are considered. Threshold Th_1 is computed, using the Otsu method [16], on the crack blocks preliminarily detected as containing crack pixels, in the training set (TrS). The second threshold value, Th_2, is computed for the blocks detected as containing crack pixels in each test set $(TstS)$ image. The lowest threshold among Th_1 and Th_2 is adopted to segment $TstS$ images as crack pixels typically present very low intensity values. This procedure is illustrated in Fig. 12 using k-means clustering for the crack detection step, followed by the elimination of isolated blocks, when applied to the image of Fig. 2(a). In this case, Th_1 takes the value of 82, and Th_2 takes the value of 75, the later being used for thresholding [see histogram in Fig. 12(b)].

Thresholding results often include pixels not belonging to existing cracks (see Fig. 13), constituting short and isolated (scattered) connected component objects (an eight

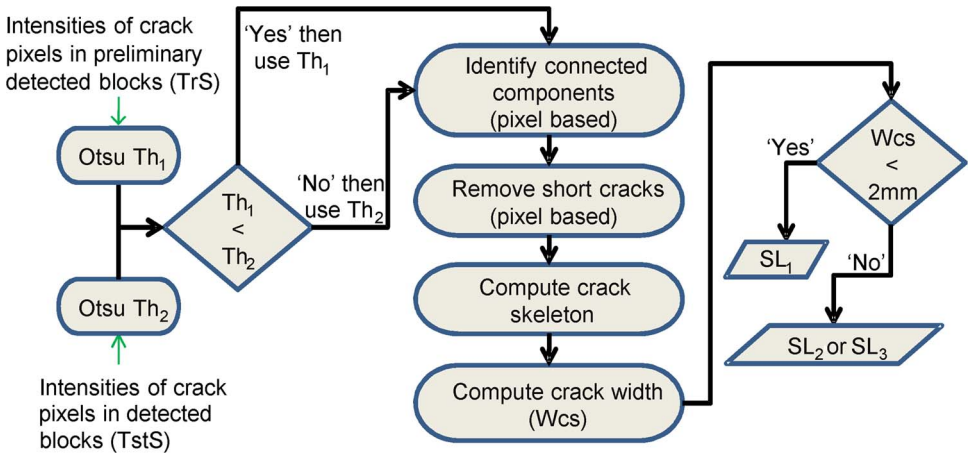

Fig. 11. Severity level assignment procedure.

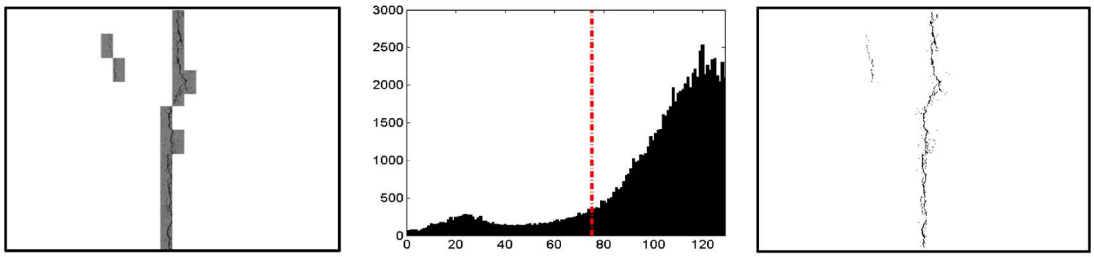

Fig. 12. (a) Crack block (normalized and saturated) intensity values for the left image of Fig. 2(a), (b) its histogram showing the threshold value used (75, i.e., Th_2 represented by the vertical dash-dot-line), and (c) the thresholded image.

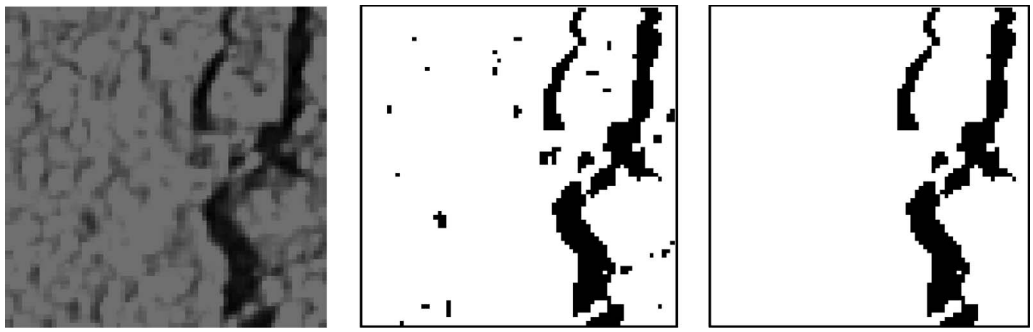

Fig. 13. (a) Image block detected as containing crack pixels in Fig. 12(a), (b) its thresholding result, and (c) the result after removing small (less than 12 crack pixels) connected components.

neighborhood has been considered), generally arising from raveling distresses [12]. Therefore, all connected components containing less than a minimum number of crack pixels (specified by the system operator) are removed, as illustrated in Fig. 13(c).

The next step is to compute the skeleton [29] of the remaining connected components, which allows calculating the average width of a crack using

$$Wcs = \frac{\text{Total number of pixels in a crack}}{\text{Total number of pixels in the crack}'\text{s skeleton}}. \quad (11)$$

As an example, for a crack connected component's area of 100 pixels having a skeleton composed by 20 pixels, the estimated crack's width would be 5 pixels. Knowing the image spatial resolution (1 pixel corresponding to 1 mm for the considered data set), this crack's average width could easily be computed as 5 mm. Fig. 14 illustrates the crack width calculation for the image in Fig. 13(c). Finally, for this example, a severity level SL_2 or SL_3 is assigned as no crack edge degradation analysis is performed at this stage.

V. Experimental Results

Performance evaluation of the proposed crack detection and characterization, with severity level assignment strategies, uses images taken during a visual pavement surface survey of a Portuguese road. The evaluation database contains 56 gray-level images for which ground truth data are available: Eight of these images contain no cracks (see samples in Fig. 15).

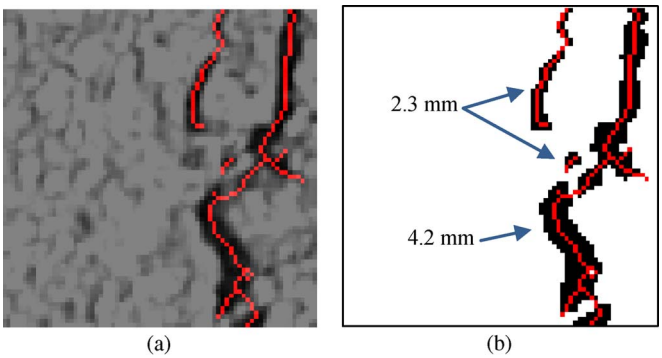

Fig. 14. Crack width measurement. (a) Identified cracks and (b) the computed width.

Part of the Matlab algorithmic implementation was supported on PRtools [25] and Ddtools [24].

As shown in Fig. 1, two types of evaluation using ground truth data, which were manually created by a skilled technician who labeled image blocks containing crack pixels and assigned a type to each identified crack, are done. The first evaluation targets the crack detection task, whereas the second addresses crack type evaluation. This two-step quantitative evaluation procedure proves to be a more robust strategy when compared with the ones commonly found in the literature.

Since currently there is no publicly available road pavement surface image database shared among research institutions, it is difficult to compare the achieved results with the published approaches. Moreover, most of the results available in the literature are only of a qualitatively nature.

The option here was to use a set of well-known metrics for quantitative evaluation purposes (a qualitative evaluation is made at the end of this section), notably *precision* (pr), accounting for the presence of false positives, *recall* (re), accounting for false negatives, and Fm (see Section III-A), together with two other global metrics: error-rate $(e - r_G)$ and crack error-rate $(e - r_{Cr})$, i.e.,

$$e-r_G = \frac{\text{Number of regions wrongly classified as classes } c_1 \text{ or } c_2}{\text{Total number of regions (cracks and no - cracks)}} \tag{12}$$

$$e-r_{Cr} = \frac{\text{Number of regions wrongly classified as class } c_1}{\text{Total number of crack regions (ground truth)}} = 1 - re. \tag{13}$$

The first set of results reported is for the preliminary image labeling of the proposed system, which aimed at an initial detection of the image blocks containing relevant crack pixels. The obtained results, with k_1 and k_2 values of 0.4 and 2.1, respectively (see Section III-A), for the 48 database images containing cracks are the following: $e - r_G = 5.0\%$; $e - r_{cr} = 36.2\%$; $re = 63.8\%$; $pr = 93.6\%$; and $Fm = 74.6\%$. These results (averaged over all images) show that the crack blocks detected at this stage effectively contain relevant crack pixels (as shown by the pr value), leading to the selection of a relevant training set (TrS).

However, its overall performance is poor (low Fm value) as a considerable number of crack blocks are not detected (as shown by the low re value), making evident the need for a more sophisticated crack detection methodology, notably to increase the recall, which is the most important metric for this type of application, where cracks should not be missed.

Using the crack detection module of CrackIT, a substantial increase in performance can be observed in Table I, the reported results being the average for all $TstS$ images. In this example, the TrS is composed of 5 images with cracks (10% of the images in the subset of the database considered), and of the remaining 51 $TstS$ images with available ground truth data, 43 contain cracks. The one-class MCDG method achieves the best performance in terms of re_{cr} (increasing from 63.8% to 98.0%), whereas the hierarchical clustering presents the best pr_{cr} value (increasing from 93.6% to 99.4%). The mixture of two Gaussian classifier presents the best overall performance of the system (Fm_{cr} increasing from 74.6% to 93.5%).

These results are considered good, notably when taking into account the difficulty of the crack detection task even for a human observer. In fact, the typical variation of human labeling leads to a result imprecision of around 1% to 2.5% due to the human ambiguity in recognizing patterns [30]. Moreover, 73% of the observed false positives and 80% of missed blocks (for the mixture of two Gaussians case) represent blocks that are neighbors of correctly detected ones, thus highlighting the mentioned crack labeling ambiguity in some situations, experienced both by the automatic algorithms, as well as by the human experts when creating the ground truth. Sample crack detection results for the images in Fig. 15, using the mixture of two Gaussian approach, are shown in Fig. 16, together with the corresponding ground truths.

For the eight $TstS$ images not containing cracks, false positives were detected only for the image shown in Fig. 15(f). In this case, the obtained results were the following: $pr_{cr} = Fm_{cr} = 0$, and *recall* is not computable.

Sample crack type characterization results are included in Fig. 17. For each connected component resulting from the crack detection stage, a crack type label is added: Longitudinal (L), Transversal (T), and Miscellaneous (M). The best crack type characterization results are obtained using the GMM clustering and the MCDG approach, as expected, since they present the best *recall* values at the crack detection task. Table II shows the results in terms of the number of undetected cracks $(e - r_{Cr})$. All the undetected cracks present a width lower than 2 mm, envisaging serious difficulties for their detection using automatic systems, as discussed in [12]. All detected cracks were correctly classified as type L, T, or M, corresponding to 100% of re_{ct}, pr_{ct}, and Fm_{ct}, respectively, which are the *recall*, *precision*, and F-measure used for crack type's evaluation (see also Fig. 1).

Since the crack detection task pr_{cr} is not 100%, a few false positives appear generally due to the small groups of dark pixels affecting two neighboring image blocks and being mistakenly considered as a short crack or due to false edges in the transition between different pavement textures together with some dark pixels in the same image area due to surface wear, as illustrated in Fig. 16(a) (bottom-row) and Fig. 16(f) (bottom-row) for short cracks.

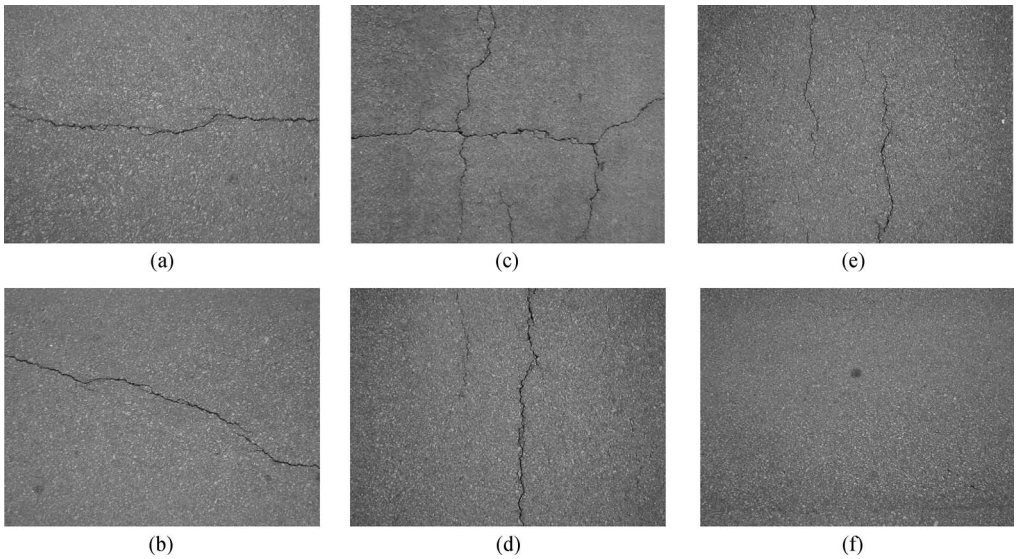

Fig. 15. Samples of road pavement database images. The bottom right image contains no crack but includes an oil spot near its center.

TABLE I
CRACK DETECTION RESULTS FOR $TstS$

Strategy	$e\text{-}r_G$	$e\text{-}r_{Cr}$	re_{cr}	pr_{cr}	Fm_{cr}
H. Clust.	1.1%	22.5%	77.5%	**99.4%**	86.0%
k-means	0.7%	12.2%	87.8%	97.6%	91.9%
Mixt. Gauss	**0.6%**	4.5%	95.5%	92.2%	**93.5%**
OC-Gauss	0.8%	13.3%	86.7%	97.1%	91.0%
OC-Min Det Gauss	1.0%	**2.0%**	**98.0%**	84.9%	90.5%
OC-Parzen	1.2%	5.0%	95.0%	84.5%	88.8%

The assignment of severity levels to the detected crack segments relies on the computed crack width values, which are listed in Table III for the images of Fig. 15. To avoid misinterpretation between raveling and short cracks with very small widths [12], crack connected components with a total area of less than 12 pixels were excluded.

A qualitative evaluation of the results presented in Table III was performed by the system operator, who infers the width of a crack by visual inspection of the corresponding images, and no inconsistencies were found in the automatically produced results. Moreover, although cracks of less than 2 mm width were not included in Table III, a few were found in other images of the database, and label SL_1 was correctly assigned.

Comparing to other published methods to detect and characterize cracks, a qualitative evaluation can be established. Multiple crack detection is also reported in [1] and [9], but CrackIT provides results of the same nature with the advantage of allowing a much more detailed evaluation, notably by computing objective evaluation metrics and by comparing against ground truth data, which were not available in those two references. In [20], although 97.4% of the images with cracks were correctly detected (only seven images out of 271 were wrongly classified), the system does not account for situations where different types of cracks are present in the same image. A direct comparison of these results with the proposed CrackIT system can only be done at the image level, in which case, CrackIT achieves 100% of correct crack detection, as every image containing cracks has been successfully identified by the proposed system.

The remaining relevant literature references provide more qualitative and global evaluation results that do not achieve better results than the ones reported here.

VI. CONCLUSION

This paper has proposed an automatic system for crack detection and characterization following the guidelines defined by the Portuguese Distress Catalog [7]. Six unsupervised crack detection approaches are confronted, with the one based on the mixture of two Gaussian models achieving the best overall performance in terms of Fm (93.5%), the best global error-rate (0.6%), and the second best performance in terms of *recall* (95.5%). These are good values, notably when an expected variation of human operator's accuracy levels between 1% and 2.5% is expected, due to human pattern recognition limitations [30].

The one-class MCDG method presents the best recall performance (98.0%), but on the other hand, its Fm and particularly *precision* results are the worst among the tested crack detection strategies.

CrackIT is able to detect multiple cracks in the same image, taking about 2 min to process the 56 images of the database for which ground truth data are available (including crack detection, type characterization, and severity level assignment), on a Pentium i5-750 processor at 2.6 GHz with 4 GB of RAM.

Crack detection and type characterization results are good, but dealing with very thin cracks (of less than 2 mm width) can be a difficult task, as many false positives may appear, notably due to the difficulty to distinguish cracking from raveling distresses. Some guidelines given by experts on automatic crack detection systems aim at the quantification of cracks of at least 3 mm width [12].

Images processed by CrackIT were acquired avoiding shadows cast by roadside objects, shrubs, trees, bridges, and viaducts, as well as those cast by the vehicle carrying the

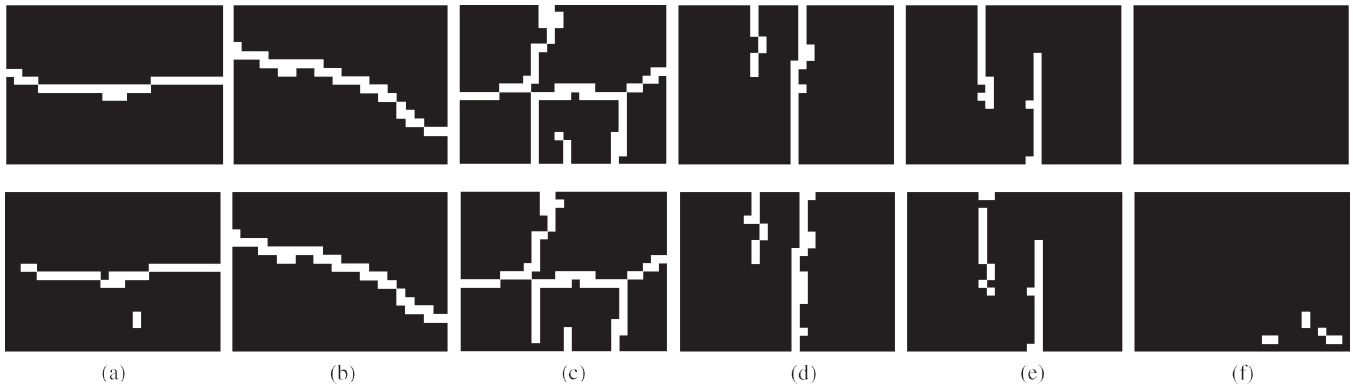

Fig. 16. Ground truth for the images shown in Fig. 15 (top row) and the corresponding CrackIT crack detection results (bottom row).

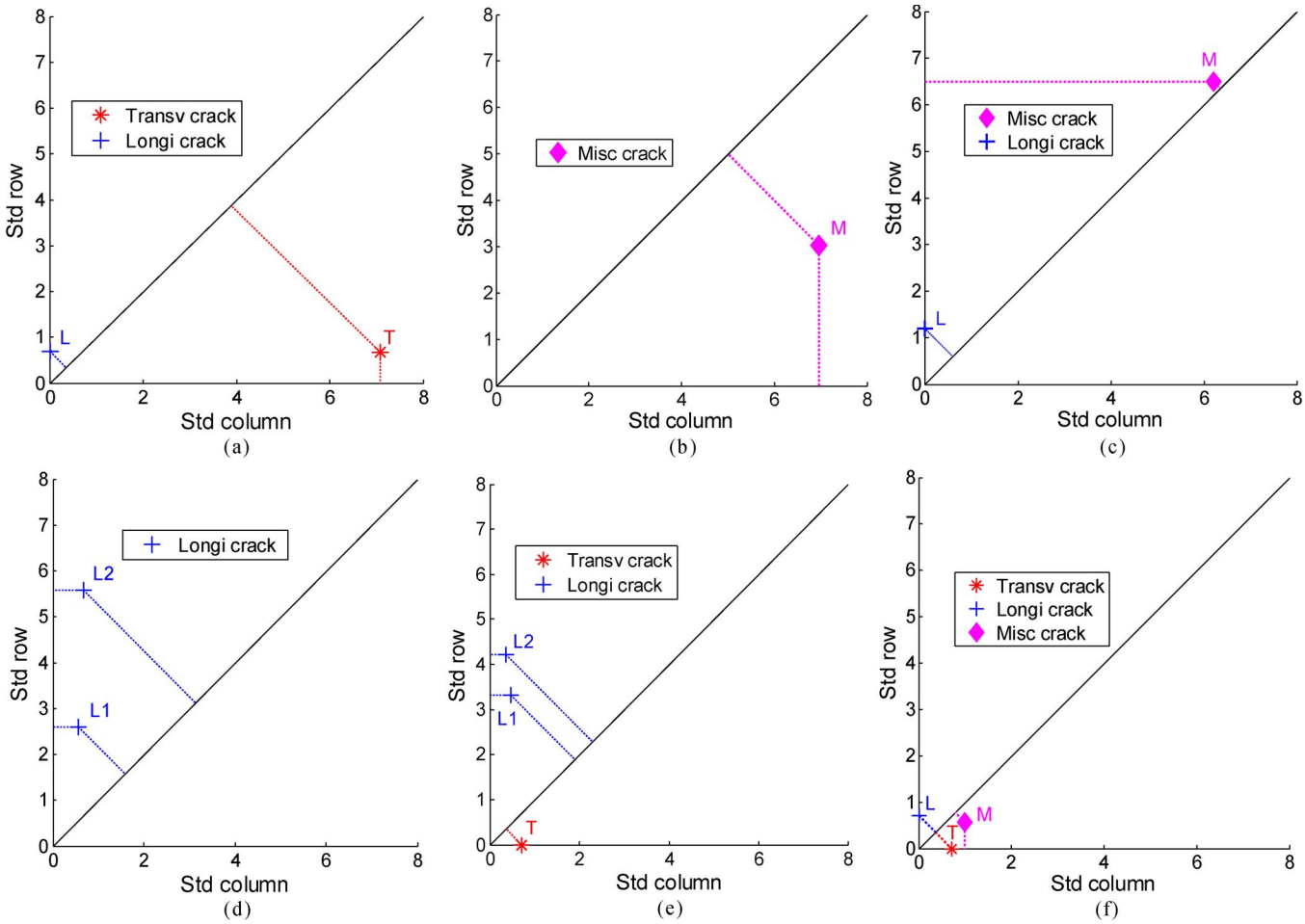

Fig. 17. Crack type characterization results for Fig. 15 images. Shorter cracks are represented nearer to the feature space origin.

TABLE II
CRACK TYPE CHARACTERIZATION RESULTS

Crack Types	Longitudinal		Transversal		Miscellaneous	
	GMM	MCDG	GMM	MCDG	GMM	MCDG
Ground truth	50	50	4	4	8	8
$e\text{-}r_{Cr}$	2	3	0	0	3	2

imaging system, mimicking the conditions obtained when using active imaging systems like INO's LRIS [31]. They were also acquired when the pavement surface was dry, since cracks in wet pavements (usually due to rainfall) are almost undetectable due to a strong loss of contrast between crack regions and the image background.

For future developments, denoising methods [32], [33] as well as entropy reduction algorithms [34], [35] should be taken into account to reduce pixel intensity variance in non-crack blocks and to increase robustness to image brightness variations. Moreover, consideration of a reject option for the classifiers, a nonuniform loss function, or the usage of one-class classifiers capable of dealing with long-tailed data distributions can be explored to improve pr when dealing with false positives

TABLE III
WIDTHS OF THE CRACK SEGMENTS DETECTED IN FIG. 14 IMAGES AND THEIR SEVERITY LEVEL ASSIGNMENTS

Image	Longitudinal	Transversal	Miscellaneous
a)	2.7 mm (SL_2 or SL_3)	3.6 mm (SL_2 or SL_3)	-
b)	-	-	4.1 mm (SL_2 or SL_3)
c)	5.2 mm (SL_2 or SL_3)	-	-
d)	left -> 2.6 mm (SL_2 or SL_3) right-> 4.2 mm (SL_2 or SL_3)	-	-
e)	left -> 3.2 mm (SL_2 or SL_3) right-> 3.6 mm (SL_2 or SL_3)	4.0 mm (SL_2 or SL_3)	-
f)	2.9 mm (SL_2 or SL_3)	3.5 mm (SL_2 or SL_3)	3.0 mm (SL_2 or SL_3)

arising from the detection task. Additionally, for crack linking purposes, the merging of crack segments using, e.g., crack width information, will be investigated.

Although the Portuguese Distress Catalog only takes into account the existence of a single label for cracks parallel to the road axis (longitudinal cracks), it is straightforward to consider additional cracking labels to comply with more complex pavement cracking definitions, as the one presented in the US LPPP protocol [36], as follows: longitudinal cracking-nonwheel path; longitudinal cracking-wheel path; fatigue cracking (usually exhibiting an alligator pattern cracking in the wheel path), edge cracking, and joint cracking. Binary masks, corresponding to the wheel path (considered as pavement areas subjected to repeated traffic loadings), centerline (near the left edge of the image), and edge (near the right edge of the image) of the road pavement, can be created and matched with the position where each longitudinal crack is detected by the proposed system.

Moreover, the system's ability to distinguish between severity levels SL_2 and SL_3 can be explored using texture analysis of the crack's boundaries. A pavement surface macrotexture analysis, as developed in [37], will be taken into account, enabling the detection of pavement surface areas presenting raveling distress, which envisages distinguishing between this type of pavement surface distress from short cracks being better. Finally, the availability of an image database shared by the automatic crack detection and characterization research community would allow a better comparison between the different approaches (the database used in this paper will be made available online).

REFERENCES

[1] S. Cafiso, A. Graziano, and S. Battiato, "Evaluation of pavement surface distress using digital image collection and analysis," in *Proc. 7th Int. Congr. Adv. Civil Eng.*, Istanbul, Turkey, 2006.

[2] Y. Huang and B. Xu, "Automatic inspection of pavement cracking distress," *J. Electron. Imag.*, vol. 15, no. 1, pp. 013017-1–013017-6, Feb. 2006.

[3] H. Cheng and M. Miyojim, "Automatic pavement distress detection system," *Inf. Sci.*, vol. 108, no. 1–4, pp. 219–240, Jul. 1998.

[4] J. Zhang, F. Wang, K. Wang, W. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1624–1639, Dec. 2011.

[5] H. Cheng and S. Hsu, "Intelligent highway traffic surveillance with self-diagnosis abilities," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1462–1472, Dec. 2011.

[6] H. Zhang and Q. Wang, "Use of artificial living system for pavement distress survey," in *Proc. 30th Annu. Conf. IEEE Ind. Electron. Soc.*, Busan, Korea, 2004, pp. 2486–2490.

[7] *Catálogo de Degradações dos Pavimentos Rodoviários Flexíveis—2nd ver., JAE, Ex-Junta Autónoma das Estradas, Almada, Portugal*, , 1997.

[8] S. Chambon, P. Subirats, and J. Dumoulin, "Introduction of a wavelet transform based on 2D matched filter in a Markov random field for fine structure extraction: Application on road crack detection," in *Proc. IST/SPIE Electron. Imag. Sci. Technol.*, San Jose, CA, 2009, pp. 72510A-1–72510A-12.

[9] Y. Sun, E. Salari, and E. Chou, "Automated pavement distress detection using advanced image processing techniques," in *Proc. IEEE Int. Conf. Electro/Inf. Technol.*, Windsor, ON, Canada, 2009, pp. 373–377.

[10] C. Ma, W. Wang, C. Zhao, F. Di, and Z. Zhu, "Pavement cracks detection based on FDWT," in *Proc. IEEE Int. Conf. CiSE*, Wuhan, China, 2009, pp. 1–4.

[11] A. Ayenu-Prah and N. Attoh-Okine, "Evaluating pavement cracks with bidimensional empirical mode decomposition," *EURASIP J. Adv. Signal Process.*, vol. 2008, no. 1, pp. 861701-1–861701-7, 2008.

[12] P. Ekdahl, *Routine Measurements of Pavement Surface Cracks*, Ramböll RST, Malmo, Sweden. [Online]. Available: http://carbon.videolectures.net/2008/contrib/surf08_portoroz/ekdahl_rmopsc/surf08_ekdahl_rmopsc_01.pdf

[13] C. Ma, C. Zaho, and Y. Hou, "Pavement distress detection based on nonsubsampled contourlet transform," in *Proc. IEEE Int. CSSE*, Wuhan, China, 2008, pp. 28–31.

[14] Q. Li and X. Liu, "Novel approach to pavement image segmentation based on neigbhboring difference histogram method," in *Proc. IEEE CISP*, Sanya, China, 2008, pp. 792–796.

[15] F. Liu, G. Xu, Y. Yang, X. Niu, and Y. Pan, "Novel approach to pavement cracking automatic detection based on segment extending," in *Proc. IEEE Int. Symp. KAM*, Wuhan, China, 2008, pp. 610–614.

[16] N. Otsu, "Threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

[17] J. Kapur, P. Sahoo, and A. Wogn, "A new method for gray-level picture thresholding using entropy of the histogram," *Comput. Vis., Graph., Imag. Process.*, vol. 29, no. 3, pp. 273–285, Mar. 1985.

[18] W. Wei and B. Liu, "Automatic road crack image preprocessing for detection and identification," in *Proc. IEEE 2nd ICINIS*, Tianjin, China, 2009, pp. 319–322.

[19] Y. Huang and Y. Tsai, "Enhanced pavement distress segmentation algorithm using dynamic programming and connected component analysis," in *Proc. 90th Meeting Transp. Res. Board*, 2011.

[20] T. Nguyen, M. Avila, and B. Stephane, "Automatic detection and classification of defect on road pavement using anisotropy measure," in *Proc. 17th EUSIPCO*, Glasgow, U.K., 2009, pp. 617–621.

[21] B. Lee and H. Lee, "Position-invariant neural network for digital pavement crack analysis," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 19, no. 2, pp. 105–118, Mar. 2004.

[22] S. Cambon and J. Moliard, "Automatic road pavement assessment with image processing: Review and comparison," *Int. J. Geophys.*, vol. 2011, pp. 989354-1–989354-20, Jun. 2011.

[23] H. Oliveira and P. Correia, "Supervised crack detection and classification in images of road pavement flexible surfaces," in *Recent Advances in Signal Processing*, A. A. Zaher, Ed. Vienna, Austria: I-Tech, 2009, pp. 159–184.

[24] D. Tax, Dec. 2009. [Online]. Available: http://prlab.tudelft.nl/david-tax/dd_tools.html

[25] R. Dwin, D. Ridder, P. Juszczak, C. Lai, P. Paclik, E. Pekalska, and D. Tax, PRTools—A Matlab toolbox for pattern recognition, PRTools, Delft, The Netherlands. [Online]. Available: http://www.prtools.org

[26] D. Tax, "One-class classification: Concept-learning in the absence of counter-examples," Delft Univ. Technol., Delft, The Netherlands, 2001.

[27] H. Oliveira and P. Correia, "Identifying and retrieving distress images from road pavement surveys," in *Proc. 15th IEEE ICIP*, San Diego, CA, 2008, pp. 57–60.

[28] A. Wright, *Developing the Automatic Measurement of Surface Condition on Local Roads*, Slovenia, 2008. [Online]. Available: http://carbon.videolectures.net/2008/contrib/surf08_portoroz/wright_dtamos/surf08_wright_dtamos_01.pdf

[29] C. Arcelli and G. Baja, "Skeletons of planar patterns," in *Topological Algorithms for Digital Image Processing*, T. Kong and A. Rosenfeld, Eds. New York: Elsevier, 1996, pp. 99–144.
[30] A. Sanz, *Pavement Surface Defects: Classification and Quantification Over a Road Network*, 2008. [Online]. Available: http://carbon.videolectures.net/2008/contrib/surf08_portoroz/amirola_psdca/surf08_amirola_psdca_01.pdf
[31] *Laser Road Imaging System (LRIS)*, INO, Quebec City, QC, Canada, Jan. 2012. [Online]. Available: http://www.ino.ca/en-ca/achievements/description/project-p/laser-road-imaging.html
[32] Q. Pan, L. Zhang, G. Dai, and H. Zhang, "Two denoising methods by wavelet transform," *IEEE Trans. Image Process.*, vol. 47, no. 12, pp. 3401–3406, Dec. 1999.
[33] L. Zhang and P. Bao, "Denoising by spatial correlation thresholding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 6, pp. 535–538, Jun. 2003.
[34] A. Suyash and R. Whitaker, "Unsupervised, information-theoretic, adaptative image filtering for image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 3, pp. 364–376, Mar. 2006.
[35] S. Kim and R. Park, "Fast local motion-compensation algorithm for video sequences with brightness variantions," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 4, pp. 289–299, Apr. 2003.
[36] *Identification Manual for the Long-Term Pavement Performance Program, Fed. Hwy. Admin., U.S. Dept. Transp., Washington, DC*, , Jan. 2012.
[37] R. Elunai, V. Chandran, and E. Gallagher, "Asphalt concrete surfaces macrotexture determination from still images," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 857–869, Sep. 2011.

Henrique Oliveira (M'10) received the degree in civil engineering and the M.Sc. degree in geographic information systems in 1995 and 2001, respectively, from the Instituto Superior Técnico, Lisbon, Portugal, where he is currently working toward the Ph.D. degree.

Since October 1995, he has been an Assistant Professor with the Engineering Department, Escola Superior de Tecnologia e Gestão–Instituto Politécnico de Beja, Beja, Portugal. Since 2003, he has been the Head of the Engineering Department, Escola Superior de Tecnologia e Gestão. His research interests include image processing, pattern recognition, geographic information systems, and remote sensing.

Paulo Lobato Correia (M'05–SM'08) received the Ph.D. degree in electrical and computer engineering from the Instituto Superior Técnico (IST), Lisbon, Portugal.

He is currently an Assistant Professor with IST, where he leads the Multimedia Signal Processing Group of Lisbon's Laboratory of Instituto de Telecomunicações (IT). He coordinates participation in several national and international research projects, dealing with image and video analysis and processing. His research interests include video analysis and processing, namely, segmentation, objective segmentation quality evaluation, content-based description, biometric recognition, and road pavement surface analysis.

Dr. Correia is an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and Elsevier's *Signal Processing*. He was Co-Chairman of the International Workshop on Image Analysis for Multimedia Interactive Services in 2004 and Program Chair for the Picture Coding Symposium in 2007. He was with EURASIP's Administrative Committee until 2008 and represents Portugal on the European Cooperation in Science and Technology Information and Communication Technologies (ICT) Domain Committee.