**CRACK DETECTION METHODS BASED ON COMPUTER VISION TECHNIQUES**

B. S. Cunha¹, C. B. Souto Maior2,\*, A. C. S. V. Negreiros3, M. C. Moura4and I. D. Lins5

**ABSTRACT**

Systems subject to continuous operation in harsh conditions are subjected to different failure mechanisms (e.g., corrosion, fatigue, and temperature related defects). In this context, inspection and health monitoring have become crucial to prevent severe damage to the system, environment, and users. However, visual inspection strongly depends on human’ experience, having its accuracy influenced by the physical and cognitive state of the inspector (i.e., human factors). Particularly, infrastructures need to be periodically inspected, which is costly, time-consuming, hazardous and biased. Nowadays, the increase in computer power allows for analysing a considerable number of images in a shorter time and use more robust algorithms. Advances in Computer Vision (CV) and Machine Learning (ML) provide the means to the development of automated, accurate, non-contact and non-destructive inspection methods. Therefore, this paper proposes adoption of different CV approaches to extract features for identification of cracks, hereby called image-based and region-based analysis. These methods were applied to a real concrete crack image database and used to feed ML models in order to identify crack existence. Results show the potential of dimensionality reduction techniques to improve methods’ performance to reach an accuracy above 94%.

*Keywords*: Infrastructure health condition, Crack detection, machine learning, computer vision, diagnosis, segmentation, texture analysis.

1. **INTRODUCTION**

Civil infrastructure elements are continuously subjected to fatigue stress and cyclic loading [1], which requires periodical inspection to ensure its safety and serviceability [2], [3] as well as to prevent long-term damage accumulation [4].

Particularly, concrete structuresare exposed to various types of damages, such as concrete cracks, steel delamination, steel corrosion, and bolt corrosion. The earliest stage of degradation is expressed in the form of surface cracks, which normally trigger maintenance actions [1], such as repair, replacement, and even evacuation [5] to avoid continuous exposure that may lead to more severe damage.

In this context, infrastructure health condition is used to monitor assets and plan maintenance policies. However, assessment is generally conducted through visual inspection by qualified human resources [1], which is costly, time-consuming, hazardous and biased [6].

Nevertheless, emerging technologies bring opportunities to support decision-makers in structure inspection, by improving the capacity, time and accuracy of inspection activities. Specifically, advances in Computer Vision (CV) and Machine Learning (ML) provide means to the development of automated, accurate, non-contact and non-destructive inspection methods[6], [7], which enables remote inspection [8].

Applications of concrete crack detection using image analysis are found in the literature mostly applied to bridges [4], [6], buildings [9] and reinforcedconcrete beams and slabs [7]. In the same way, although pavements are exposed to different kinds of tensions caused by traffic loads and temperature oscillation, pavement cracks are especially critical since they affect both vehicles (speed, wear out) and safety [10].

According to [1], difficulties in image crack detection are mostly based on the irregular shape and size of cracks, noisy background, quality of the image (i.e. resolution, illumination, shading, low contrast, reflection) and presence of irrelevant objects (e.g. oil spots, road signs) [10], [11]. Hence, a suitable alternative to deal with these issues is the usage of image pre-processing and segmentation techniques.

For instance, [10] pre-process images applying intensity correction, irrelevant object elimination and crack saliency using an extension of grid cell analysis. As for segmentation, crack extraction is based on spatial distinctiveness, adaptive threshold segmentation, and further post-processing operations to fix discontinuities. Other examples of threshold segmentation are also found in [12], [13]. [6], [14] applied classical edge detection techniques to segment crack contours from the background, while a morphology-based segmentation is found in [15].

Morphological image pre-processing is also appropriate for crack extraction, however, limitations of the standalone approach normally require combination with other techniques to improve performance [10]. In the context of infrastructure assessment, [16] proposed a crack detection and thickness quantification methodology based on 3D scene reconstruction, morphological image processing, feature extraction and pattern recognition using distinct ML techniques, such as Support Vector Machine (SVM), Neural Network (NN) and Nearest-Neighbour. In the same context, [7], [11] presented a methodology for crack identification based on segmentation, feature extraction, and classification also through ML techniques, such as Multilayer Perceptron (MLP) and SVM. Although this paper follows similar structure as the previous ones, we extended the scope to consider different feature extraction methods.

For instance, another classical technique used in CV and image processing is texture classification (TC). According to [17], texture analysis has been an area of intense research, applied in areas such as medical image analysis, remote sensing, object recognition, document analysis, and environment modeling.Nowadays, TC is also commonly found in areas such as system reliability and fault detection. For instance, [5], [18] proposed a method to retrieve the crack contour through texture analysis. [19] combine the use of Hough transform features and Local Binary Pattern (LBP) to extract edge orientation and texture features, building crack seeds, which are further processed in order to obtain a clearer image of the crack. Although TC is widely applied in different contexts, there is still a need to enhance its effectiveness for crack detection in images with high levels of surface texture [1].

Conversely to the above-mentioned papers that deals with the classification of individual segmented objects from the original image, [4] used Convolutional Neural Network (CNN) for inspection of concrete constructions by detecting whether the image contains or not crack(s). They verified that image quality is an essential factor in detection accuracy and emphasise the need to diversify defects in the training dataset. [8] proposed an approach for quasi-real-time simultaneous detection of multiple types of concrete damage, using a Faster Region-based CNN (Faster R-CNN). The advantage of this approach is the possibility of detecting several types of damages, but it entails low precision.

Although Deep Learning (DL) techniques, such as CNN, have increasingly been applied in a wide range of areas [20], its performance heavily depends on the architecture applied and on the size and variety of the training dataset [4]. Because of that, DL techniques are computationally intensive [21], requiring the use of GPUs and limiting its application in standard computers. Moreover, aside from the fact that a large data set can be proven difficult to obtain in some cases, it requires a lot of manual effort for labelling [22].

On the other hand, with approaches based on image pre-processing, segmentation and classification, it is possible to achieve desirable performance results even for small data sets. So, this paper aims at developing a robust methodology for crack detection that addresses the limitations of (i) not having big datasets, (ii) standard computational resources, and (iii) poor crack image quality, i.e., noise on the image and from the background, overcoming the previously mentioned limitations.

In this way, this paper compares different approaches to extract features and identify cracks in concrete images. Segmentation and TC methods, here addressed as region-based and texture-based analysis respectively, are used to extract features from the images and feed ML models, i.e., SVM and MLP, in order to detect crack existence. The features implemented for each approach are based on the existing literature and variants of classical approaches new to this application. Hence, the idea is to determine which features are relevant for crack detection to improve computational speed. Furthermore, the proposed methodology aims at overcoming the problem caused by the variety on both crack and background characteristics by determining appropriated and discriminative features. Therefore, the proposed methodology will be tested in a recently available concrete crack data base, with diverse image characteristics.

The remainder of this paper is organized as follows. Section 2 describes the applied methodology and specific concepts regarding CV techniques when dealing with crack detection. Section 3 presents the crack-related database used to evaluate the proposed model as well as the results of this application. Section 4 summarizes the achievements and concludes remarks.

1. **METHODOLOGY**

Based on images, the proposed methodology relies on extracting features to be used as input to ML techniques for crack detection. The image in our case is a segmented version of the original one. Particularly, the segmentation aims to create a new image containing only white parts that resembles a crack (the procedure will be described in details in the following section). In this paper, we analyzed two approaches for automated crack detection separately, extracting different features in each case, and comparing their performance in labeling an image as cracked or non-cracked.

The difference between them relies on the focus of analysis, as one analyses the whole image, and the other analyses each region of the segmented image (i.e., connected white pixels). The former is the texture-based analysis and the latter is the region-based analysis, which can also localize cracks in the image. Note that, the problem complexity increases from the texture-based to the region-based analysis, as the result gets more detailed by changing the focus of analysis.

For example, in “Figure 1”, the texture-based approach would indicate that this image contains crack(s), i.e., label this image as cracked. With this result, it is not possible to know how many cracks exists (in this case, two). The region-based approach, on the other hand, attempts to indicate the exact pixels that define each crack, and hence label this image as cracked.

Uma imagem contendo onda, surfando, cavalgando, ao ar livre

Descrição gerada automaticamente

**Fig. 1**. Example of a concrete crack image containing two cracks, obtained from ÖZGENEL (2018).

Then, we also considered possible application of two dimensionality reduction techniques, i.e., Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA). This step aims at reducing the number of features used to describe the image, making the model more efficient and robust. The proposed methodology is summarized in “Figure 2” and detailed in the next sections.

Uma imagem contendo texto

Descrição gerada automaticamente

**Fig. 2**. The methodology proposed for crack detection.

2.1 Segmentation

Segmentation comprises the identification of objects that can potentially be classified as cracks in the image, removing irrelevant or unrelated information that belongs to concrete pattern characteristics (i.e., noise, background). The segmentation technique was implemented in MATLAB and is based on the morphological operation proposed by [16], shown in “Eq. (1)”.

|  |  |
| --- | --- |
|  | (1) |

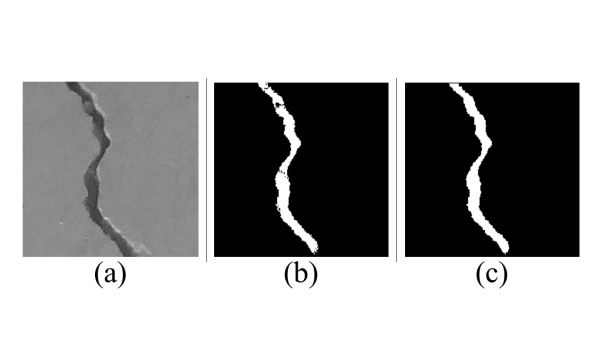
where Iis the grayscale image, ○ is the opening morphological operation, ● is the closing morphological operation, and SE is the linear structuring element, defined for orientations of 0⁰, 45⁰, 90⁰ and 135⁰.

It is essential to understand that the structuring element is determined by two parameters: format and size. The linear format chosen for SE enables the segmentation of objects perpendicular to each orientation, while the size adopted narrows the segmentation by imposing a constraint for object thickness. Therefore, for each image, the morphological operation described in “Eq. (1)” is repeated for several SE of different sizes (from 2 to 35) in order to map the most information about objects. The resulting image of each iteration is binarized by Otsu’s method [23], filtered by the area of the segmented object and joined together to compose one single image.

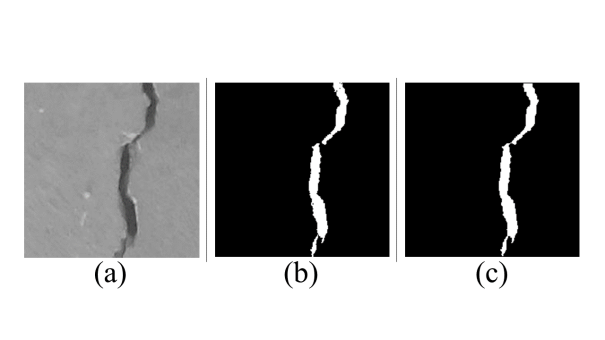
“Figure 3(b)” and “Figure 4(b)” present examples of the segmentation result above described, yet still containing noise inside the crack and related to the background (i.e. black dots and/or discontinuity). In order to address this issue, the following steps, based on the method proposed by [11], were used to fill internal discontinuity and eliminate broken parts of the crack.

1. Fill holes in the segmented crack image;
2. Perform morphological closing (dilation followed by erosion);
3. Obtain the skeleton line of the crack to identify crack direction;
4. Search for breaking points of the crack and connect points that are at most 5 pixels away from each other in the direction of the skeleton line.

Steps (iii) and (iv) were implemented using methods of “Edge Linking and Line Segment Fitting” developed by (Kovesi, 2000) and results are shown in “Figure 3(c)” and “Figure 4(c)”.



**Fig. 3**. (a) grayscale image, (b) segmented image containing holes, (c) segmented image filled.



**Fig. 4.** (a) grayscale image, (b) segmented image containing broken parts of the same crack, (c) segmented image with crack parts connected.

2.2 Feature extraction

Features are quantitative attributes or properties that aside from describing characteristics of the image (or image regions) also help to distinguish different patterns [16]. In this context, it is essential that features are invariant to changes in condition, such as illumination for example (Nixon and Aguado, 2008). A feature extraction process is a particular form of extracting information from the image for comparison and analysis purposes [7].

In this paper, we analyzed the texture- and region-based approaches for feature extraction, which were implemented in MATLAB and are described in the following sections.

2.2.1 Texture-based analysis

The first approach considered in this paper extracts features from the entire image (i.e., textural properties). According to (Nixon and Aguado, 2008), there is not a precise definition for texture, as it is usually attributed to human perception, and then the concepts for its analysis are flexible enough to handle different definitions. In this paper, we are interested in distinct textures presented in an image with a crack, once it is considerable different from concrete image without any crack.

A classical and widely used method to describe textures is the Local Binary Pattern (LBP), which was initially proposed by Ojala, Pietikäinen, and Harwood (1996), and consists of creating a histogram of textural information by determining the relationship between each pixel and its neighbourhood. The LBP presents advantages such as a notable computational efﬁciency and a good texture discriminative property. However, limitations such as sensibility to noise and image rotation, small spatial support, and loss of local textural information motivate improvements on the method.

[17] proposed an improvement considering two different types of features in a local patch: the pixel intensities and the pixel differences. The pixel intensitiesare divided into two components: the intensity of the central pixel (CI) and the intensities of its neighboring pixels (NI). The CI descriptor consider the relation between the intensity of the central pixel of the whole image compared to the other intensity pixels. The NI descriptor, differently from the CI, compare the intensity of the central pixel of the path with the entire path neighbouring pixels. For the pixel difference, the radial difference (RD) is analyzed. The RD descriptor consider a radial integer displacement to compute the difference of intensity between the central pixel of the path and their neighbourhood. For more details see[17].

In this paper, we applied LBP and its three above presented variants in the segmented image, generating four different histograms for each image. For each image, we extracted classical features (i.e., contrast, correlation, energy, and homogeneity) from each of its four histograms (generating 16 features) and also other 21 features from the image itself (Table 1), considering recently proposed features by [7]. The outcome of this step is a feature vector composed by 37 features, which are described in “Table 1”.

**Tab. 1**. Description of textural properties.

|  |  |
| --- | --- |
| Feature | Description |
| Contrast | Intensity contrast between a pixel and its neighbor over the whole image. |
| Correlation | Correlation between a pixel and its neighbor over the whole image. |
| Energy | Sum of squared elements in the Gray Level Co-occurrence Matrix (GLCM) |
| Homogeneity | Closeness between the distribution of elements in GLCM its diagonal. |
| Sum of Major Axis Lengths | Sum of the length of the major axis of the ellipse that has the same normalized second central moments as the crack. |
| Crack Length | Sum of individual crack lengths. |
| Perimeter | Sum of individual crack perimeters. |
| Average Major Axis Length | Average length of the major axis of the ellipse that has the same normalized second central moments as the crack. |
| Aspect Ratio | Average ratio of Major Axis Length over Minor Axis Length. |
| Perimeter | Average crack perimeter. |
| Compactness | Average ratio of Perimeter over Crack Length. |
| Thresh Out | Relative proportion of the cracks to non-crack objects. |
| Entropy | Statistical measure of randomness. |
| Local variance | Average local standard deviation of 3 × 3 neighborhood around each pixel in the image. |
| Euler Number | Total number of cracks in image minus the total number of holes in those cracks. |
| Standard Deviation | Standard deviation of all values. |
| Ixx | Moments of the area of the pixels about the centroidal x-axis. |
| Iyy | Moments of the area of the pixels about the centroidal y-axis. |
| Ixy | Moments of the area of the pixels about the centroidal x- and y-axes. |
| Polar Moment | Sum of Ixx and Iyy. |
| Total Distance Between Cracks | Sum of the distance between cracks. |

2.2.2 Region-based analysis

Unlike texture-based approach, region-based analysis concerned with parts of the image. In this way, the latter focuses on the extraction of features for each segmented region of the image. Here, a region is defined as connected white pixels that represent the crack.

In this paper, the feature vector considered for each region includes features provided in [11], [16]. All 13 features are described in “Table 2” and should capture the characteristics of the crack that distinguishes it from the background and from other objects. By classifying segmented regions, it is possible to determine the number of cracks in the image and the precise location of each crack.

**Tab. 2**. Description of region properties.

|  |  |
| --- | --- |
| Feature | Description |
| Eccentricity | Ratio between the distance of the ellipse focusthat has the same second-moments as the region and its major axis length. |
| Area ratio | Ratio between the area of the region and the area of the ellipse that has the same second-moments as the region. |
| Solidity | Proportion of the pixels in the convex hull thatis also in the region. |
| Compactness | Ratio between the square root of the  region and its perimeter. |
| Convex Area | Number of pixels in the image that specifies the convex hull. |
| Equivalent Diameter | Diameter of a circle with the same area as the region. |
| Euler Number | Number of objects in the region minus the number of holes in those objects. |
| Filled Area | Number of pixels in the region with filled holes. |
| Major Axis Length | Length (in pixels) of the major axis of the ellipse that has the same normalized second central moments as the region. |
| Perimeter | Distance (in pixels) around the boundary of the region. |
| Length-width ratio | Length-width ratio of the smallest circumscribed rectangle of the crack. |
| Proportion of crack pixels | Ratio of the area of the region and the area of the smallest circumscribed rectangle of the crack. |
| Image Concentration (K) |  |

2.3 Dimensionality reduction

The aim of this step is to reduce the number of variables for speed and robustness purposes. We used two kinds of methods to reduce the problem dimensionality: LDA and PCA. Even though this step is optional, it generally provides more accurate which worth investigation. Then, we also compared the classification results with and without dimensionality reduction technique.

LDA is a widespread supervised classifier with a linear decision boundary which is also applied to dimensionality reduction. It reduces data to the desired size (as long as it is less or equal to the number of categories minus one) by maximizing the ratio of the between-class distance to the within-class distance [24]. It also projects the input in the most discriminative directions, been considerable simple and mathematically robust. In our case, since we are interested in only two categories (i.e., with and without crack), the application of LDA reduces the feature vectors to just one element.

Alternatively, PCA is an unsupervised method that aims to project the dataset into a lower dimensional space (the principal components) with an orthogonal linear transformation and preserve as much ‘variability’ as possible [25]. PCA defines how many and which features are considered relevant for image representation. As LDA reduces the feature vector dimensionality to one, we set PCA to reduce it to one. Moreover, we also considered models without any dimensionality reduction and compare results.

2.4 Classification

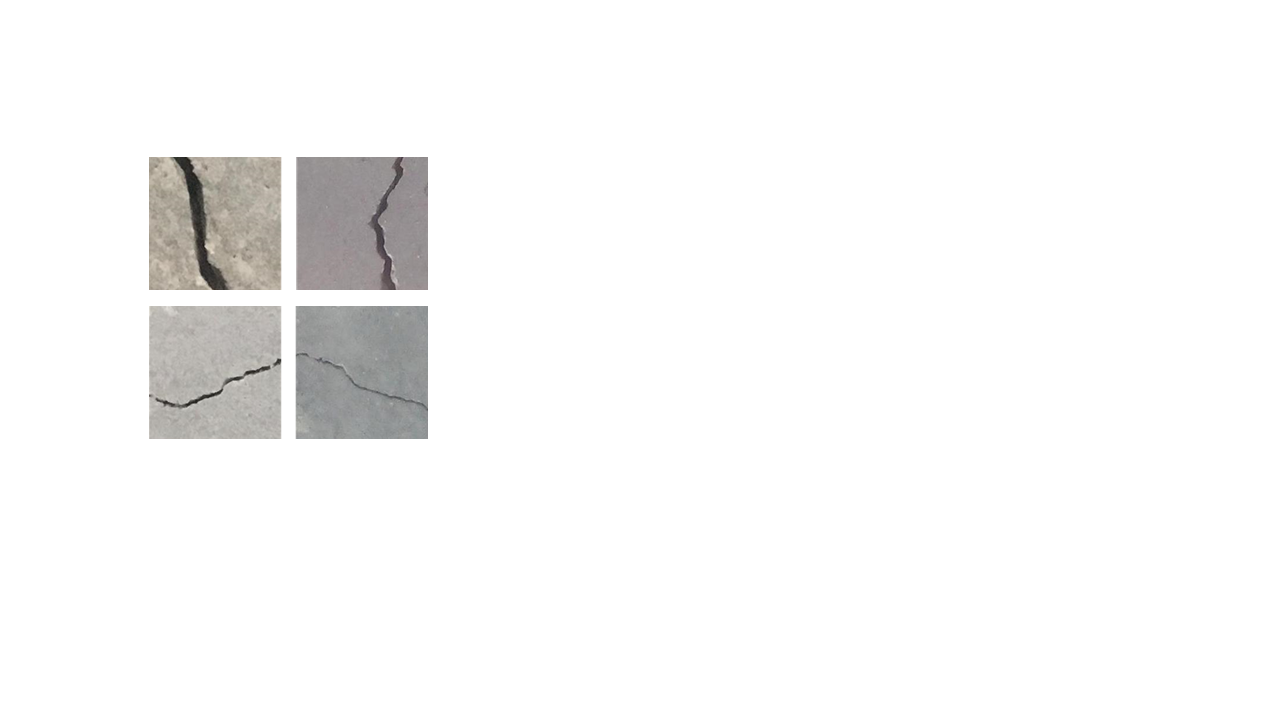
The next step after the dimensionality reduction is the classification (see Figure 2). To that end, we choose two popular ML models: MLP and SVM, comparing results provided by both of them. Both SVM and MLP were trained for texture-based and region-based features along with the ground truth, resulting in twelve different models.

For SVM, we used as kernel function the Radial Basis Function (RBF). MLP was implemented applying one hidden layer with 100 hidden neurons and a single output layer. The activation function was the hyperbolic tangent (tanh) function and the number of epochs was 200.

Note that due to the dimensionality reduction step, the feature vector for each case might not be composed of all the features described in “Table 1” and “Table 2”. Results are presented in the next section, comparing the performance of the classification models with the texture- and region-based features, considering the cases with the implementation of PCA and LDA against case with no dimensionality reduction. For this step, we used the Python computational language and the Google Collaboratory, which is a free cloud platform for Python (2.7 and 3.6).

1. **DISCUSSION**

In this paper, we used a database that contains 458 concrete crack images, with resolution 4,032 x 3,024 pixels, taken from the Technical University of the Middle East (METU) buildings Özgenel (2018). These images were divided smaller images, with 227 x 227 pixels, each of them representing concrete parts of these buildings containing or not cracks. From these images, a total of 1,768 crack images and 1,246 non-crack images were selected. Examples of the data set are shown in “Figure 5” and “Figure 6”, demonstrating the variety in both cracking and concrete characteristics.



**Fig. 5**. Examples of crack images.



**Fig. 6**. Examples of non-crack images.

With this data set, a comparison between texture- and region-based analysis was performed using SVM and MLP. For each case, 80% of the feature vectors were used for training purposes. The main results are shown in “Table 4”.

Performance metrics, i.e., accuracy andsensitivity were calculated [16]. The accuracy of each method is presented in “Table 4” with the best ones highlighted in boldfaced and the worst ones. Accuracy measures the proportion of correct classifications. Sensitivity, on the other hand, is the ability of the classifier to find all the positive samples correctly and, hence, is an informative measure when there is a high cost associated with false negatives. In reliability analysis, in general, sensitivity is the most critical metric once the cost of a false negative can be catastrophic if, for instance, human safety is involved. The sensitivity results are shown in “Figure 7” and “Figure 8” only for the best region-based and texture-based models, respectively.

Note that the result of the texture-based analysis is the classification of the image as crack or non-cracked, while the outcome of the region-based approach is the classification of each segmented region as a crack or not. For instance, if a given image presents more than one crack, and if the region-based approach correctly classifies at least one of them, it is possible to classify the corresponding image as cracked. Then, the performance metrics discussed above were calculated based on the classification for the whole image in order to have a common ground for models’ comparison.

“Table 3” shows that the implementation of LDA considerably improved the classification performance, while the worst case is when no dimensionality reduction is considered, which is almost a guess since it implies a fifty percent chance of correctly classifying the observation. For the region-based approach, both SVM and MLP reached 94% accuracy using LDA. As for the texture-based, the best performance was obtained with MLP and LDA again, reaching 98% accuracy. In fact, this result surpassed all the others.

**Tab. 3**. Classification performance in terms of accuracy, per dimensionality reduction method.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method | Classification | None | PCA | LDA |
| Texture-based | SVM | 0.60 | 0.60 | 0.97 |
| MLP | 0.75 | 0.76 | **0.98** |
| Region-based | SVM | 0.50 | 0.75 | 0.94 |
| MLP | 0.91 | 0.86 | 0.94 |

From “Table 3”, one can also note that results for the texture-based analysis are generally superior to the ones for the region-based. This result is somewhat expected since the former is concerned with labeling the whole image as with a crack or not, while the latter aims at locating the crack on the image, increasing the problem complexity. However, especially for infrastructures, which have a large area to be covered, one can take advantages with the combination of both approaches, as the first approach can be used to identify areas of interest, where there might be the presence of cracks. Hence, the second can narrow down the precise location of the cracks.

From “Figure 7” and “Figure 8”, it is possible to note that the main improvement between these approaches is related to the sensitivity, from 93% in the region-based to 98% in the texture-based. As pointed out before, in the context of reliability analysis, the cost of false negatives can be very high, and thus sensitivity is a useful metric for the model evaluation.

Uma imagem contendo captura de tela

Descrição gerada automaticamente

**Fig. 7**. Confusion matrix for the best region-based model.

Uma imagem contendo captura de tela

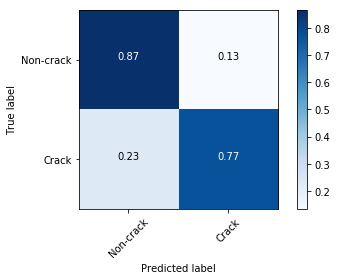
Descrição gerada automaticamente

Fig. 8. Confusion matrix for the best texture-based model.

100 images from SDNET (30 non-crack and 70 crack).

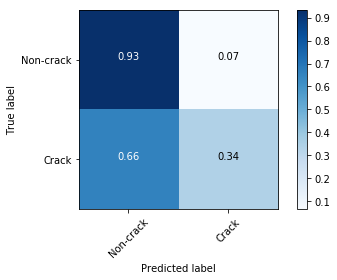
* Region-based: MLP+LDA

Accuracy = 77%



* Texture-based: MLP+LDA

Accuracy = 52%



1. **CONCLUSION**

In order to create an automatic concrete crack detection model, two different approaches were proposed in this paper. The first one was based on segmentation techniques, while the second one relies on texture analysis. Even though using a pre-processing technique for dimensionality reduction is optional for each of these approaches, both have to be combined to a classification step to assess the existence of crack, in which ML models (i.e., SVM and MLP) were required, allowing for a comparison between these learning techniques. A real concrete crack image data base was usedto evaluate performances.

As results, models with LDA dimensionality reduction and segmentation (i.e., texture-based) approach generally presented better results. Moreover, models based on MLP learning techniques presented improved results compared with SVM-based ones.

Note that although we are analyzing the approaches mentioned above separately, for comparison purposes, our ongoing research focuses on combining them. Combining both approaches could be interesting when dealing with a large area to be covered by inspection, in which case the area could be dividedinto several parts. For each part, one could use the texture-based approach to detect whether there is a crack or not and then for each positive result, use the region-based analysis to narrow down the precise location of the crack(s).

As a future work, we expect to improve the image detection methodology to classify not only cracks, but also other types of fault (e.g., corrosion, deformation, wear). Moreover, in possession of a considerable amount of images, DL approaches can be applied to this context, and compared with our results in terms performance metrics and required computational power.

1. **ACKNOWLEDGMENT**

Thisstudywasfinanced in partbythe Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - FinanceCode 001. The authors also thank the Brazilian research funding agencies CNPq and FACEPE for financial support through research grants.

1. **REFERENCES:**

[1] A. Mohan and S. Poobal, “Crack detection using image processing : A critical review and analysis,” *Alexandria Eng. J.*, vol. 57, pp. 787–798, 2018.

[2] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, “Advanced Engineering Informatics A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure q,” *Adv. Eng. Informatics*, vol. 29, pp. 196–210, 2015.

[3] S. Sankarasrinivasan, E. Balasubramanian, K. Karthik, and U. Chandrasekar, “Health Monitoring of Civil Structures with Integrated UAV and Image Processing System,” *Procedia Comput. Sci.*, vol. 54, pp. 508–515, 2015.

[4] S. Dorafshan, R. J. Thomas, and M. Maguire, “Deep Learning Neural Networks for sUAS-Assisted Structural Inspections : Feasibility and Application,” *2018 Int. Conf. Unmanned Aircr. Syst.*, pp. 874–882, 2018.

[5] M. Ruiz *et al.*, “Wind turbine fault detection and classification by means of image texture analysis,” *Mech. Syst. Signal Process.*, vol. 107, pp. 149–167, 2018.

[6] S. Dorafshan, M. Maguire, N. V. Hoffer, and C. Coopmans, “Challenges in Bridge Inspection Using Small Unmanned Aerial Systems : Results and Lessons Learned,” in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2017, pp. 1722–1730.

[7] R. Davoudi, G. R. Miller, and J. N. Kutz, “Structural Load Estimation Using Machine Vision and Surface Crack Patterns for Shear-Critical RC Beams and Slabs,” *J. Comput. Civ. Eng.*, 2018.

[8] Y. Cha, W. Choi, G. Suh, and S. Mahmoudkhani, “Autonomous Structural Visual Inspection Using Region-Based Deep Learning for Detecting Multiple Damage Types,” vol. 33, pp. 731–747, 2018.

[9] S. Dorafshan and M. Maguire, “Automatic Surface Crack Detection in Concrete Structures Using OTSU Thresholding and Morphological Operations,” 2016.

[10] D. Zhang, Q. Li, Y. Chen, M. Cao, L. He, and B. Zhang, “An efficient and reliable coarse-to-fine approach for asphalt pavement crack detection ଝ,” vol. 57, pp. 130–146, 2017.

[11] S. U. N. Liang, X. Jianchun, and Z. Xun, “An Algorithm for Concrete Crack Extraction and Identification Based on Machine Vision,” *IEEE Access*, vol. 6, pp. 28993–29002, 2018.

[12] Y. Fujita and Y. Hamamoto, “A robust automatic crack detection method from noisy concrete surfaces,” pp. 245–254, 2011.

[13] Z. Yiyang, “The Design of Glass Crack Detection System Based on Image Preprocessing Technology,” *2014 IEEE 7th Jt. Int. Inf. Technol. Artif. Intell. Conf.*, pp. 39–42, 2014.

[14] F. C. Pereira, C. E. Pereira, F. C. Pereira, C. E. Pereira, and C. Eduardo, “ScienceDirect Recognition of Cracks using UAVs Embedded Recognition of Cracks using UAVs for for Automatic Automatic Recognition Recognition of of Cracks Cracks using using UAVs UAVs,” pp. 16–21, 2015.

[15] S. Y. Alam, A. Loukili, F. Grondin, and E. Rozière, “Use of the digital image correlation and acoustic emission technique to study the effect of structural size on cracking of reinforced concrete,” *Eng. Fract. Mech.*, vol. 143, pp. 17–31, 2015.

[16] M. R. Jahanshahi, S. F. Masri, C. W. Padgett, and G. S. Sukhatme, “An innovative methodology for detection and quantification of cracks through incorporation of depth perception,” *Mach. Vis. Appl.*, 2013.

[17] L. Liu, L. Zhao, Y. Long, G. Kuang, and P. Fieguth, “Extended local binary patterns for texture classification,” *Image Vis. Comput.*, vol. 30, no. 2, pp. 86–99, 2012.

[18] F. A. Fardo, G. H. B. Donato, and P. S. Rodrigues, “Texture Analysis for Crack Detection in Fracture Mechanics,” *J. Fail. Anal. Prev.*, vol. 18, pp. 526–537, 2018.

[19] M. Quintana, J. Torres, and J. M. Menéndez, “A Simplified Computer Vision System for Road Surface Inspection and Maintenance,” vol. 17, no. 3, pp. 608–619, 2016.

[20] H. Wang, G. Li, G. Wang, J. Peng, H. Jiang, and Y. Liu, “Deep learning based ensemble approach for probabilistic wind power forecasting,” *Appl. Energy*, vol. 188, pp. 56–70, 2017.

[21] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, “Deep Learning and Its Applications to Machine Health Monitoring: A Survey,” vol. 115, pp. 213–237, 2016.

[22] B. E. Mneymneh, M. Abbas, and H. Khoury, “Evaluation of computer vision techniques for automated hardhat detection in indoor construction safety applications,” *Front. Eng. Manag.*, vol. 5, no. 2, pp. 227–239, 2018.

[23] N. Otsu, “A Threshold Selection Method from Gray-Level Histograms,” *IEEE Trans. Syst. Man. Cybern.*, 1979.

[24] J. Ye, R. Janardan, and Q. Li, “Two-Dimensional Linear Discriminant Analysis,” in *18th Annual Conference on Neural Information Processing Systems (NIPS 2004)*, 2004.

[25] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. Springer-Verlag, 2002.