

Automated segmentation of concrete images into microstructures: A comparative study

Mehran Yazdi* and Katayoon Sarafrazi

¹Department of Electronics and Computer Engineering, Shiraz University, Shiraz, Iran

(Received December 14, 2013, Revised May 15, 2014, Accepted July 13, 2014)

Abstract. Concrete is an important material in most of civil constructions. Many properties of concrete can be determined through analysis of concrete images. Image segmentation is the first step for the most of these analyses. An automated system for segmentation of concrete images into microstructures using texture analysis is proposed. The performance of five different classifiers has been evaluated and the results show that using an Artificial Neural Network classifier is the best choice for an automatic image segmentation of concrete.

Keywords: microstructural analysis; image segmentation; FLD; KNN; artificial neural networks; SVM; bayesian classification; co-occurrence matrix; texture analysis

1. Introduction

Concrete mixtures are complex heterogeneous materials composed of air voids, cement pastes, and aggregates. The overall performance and properties of concrete depend on the proportion and distribution of these materials. However, the microstructural analysis of concrete images is a time consuming process which is done manually or using semi-automatic methods.

Recently, numerous studies have been done on performing an automatic analysis of concrete images. We can mention some of these works as follows; the investigation of micro-cracks and voids which is a useful tool for extracting crack patterns and damage characterization (Ammouche, *et al.* 2001; Bernstone and Heyden 2009; Carloni and Subramaniam 2013; Chen *et al.* 2011; Glinicki and Litorowicz 2006; Litorowicz 2006; Martins *et al.* 2013; Ringot and Bascoul 2001; Shamshad *et al.* 2013; Soroushian *et al.* 2003) or segmentation of different phases (separation of aggregate particles from paste matrix) in cement paste based materials (Dequiedt *et al.* 2001; Silva *et al.* 2002; Yang and Buenfeld 2001) or some other works (Dequiedt *et al.* 2013; Feng *et al.* 2004; Liang *et al.* 2009; Redon *et al.* 1999; Soroushian and Elzafraney 2005; Wong *et al.* 2006). For the purpose of atomization, different segmentation methods have been recently proposed. Yang (Yang and Buenfeld 2001) used a combination of binary operations such as binary edge detection, thresholding, opening and erosion to automatically segment concrete images. Dequiedt (Dequiedt

*Corresponding author, Associate Professor, E-mail: yazdi@shirazu.ac.ir

^aStructural Engineer, E-mail: sarafrazi@shirazu.ac.ir

et al. 2001) used different color spaces to separate different phases in concrete images. Silva (Silva *et al.* 2002) used morphological analysis for segmentation. Stochastic methods such as Markov Random Fields were also used (Liang *et al.* 2009). Indeed, concrete images consist of many texture regions and there have not been enough studies based on textural nature of concrete images.

In recent decades, texture analysis has been one of the hottest research topics in image processing field. Many algorithms have been developed and used for analyzing and synthesizing texture (Petrou *et al.* 2006; Tuceryan and Jain 1993). A textured region may consist of many small homogenous regions (texture elements) and therefore segmenting textured images using traditional segmentation methods results in over-segmentation. The goal of texture segmentation is to group

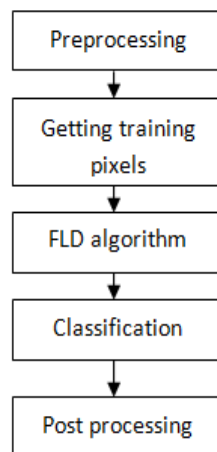


Fig. 1: An overview of the proposed segmentation method



Fig. 2 Examples of test images

image pixels into regions with similar texture. A texture analysis method uses values of neighboring pixels to represent texture (Sarafrazi *et al.* 2013). In this study, grey-level co-occurrence matrixes (GLCM) have been used for texture analysis. GLCM is a second-order statistics method which can be used to extract texture's features.

In this paper, a new method for automatic segmentation of concrete images into different phases using texture analysis is proposed. In the phase of selecting a classifier, different classification methods can be used. We also compare the performance of most utilizing methods in order to determine the best one.

In the next section, the proposed method and its different stages are presented.

2. Proposed method

Fig. 1 shows the flowchart of the proposed method. Each step of the algorithm will be explained in the following sections.

2.1 Image acquisition

In our work, a dataset of 31 images of concrete with 2 mm resolution were used. We used 20 images for training procedure and the rest for evaluation part. Fig. 2 shows some of these test images.

2.2 Grey-level co-occurrence matrix (Haralick, Shanmugam and Dinstein 1973)

GLCM is a second-order statistics method which can be used to extract texture's features. The GLCM can model the texture by calculating how often a pair of pixels with specific values and in a specified spatial relationship occurs in an image. Different statistical measures can be extracted from GLCM and used to model texture.

To form the GLCM for each pixel an 25×25 window around a given pixel is used as neighboring pixels. Then the GLCM is calculated for the pixels in this window.

Different spatial relationships can be used to form a GLCM. In this study, this relationship is defined as between the pixel of interest and the pixel to its immediate right side (i.e., horizontally adjacent). Since each pixel can have any integer values between 0 and 255, the size of GLCM would be 256×256 (for each pixel). In order to reduce the data volume, pixel values are quantized to 8 levels (i.e., 0, 32, 64, 96, 128, 160, 192 and 224). For example, all pixels with values between 64 and 95 are quantized to 64. This would reduce the GLCM size to 8×8 pixels.

After the GLCM is formed using different statistics which describe texture can be extracted from it. In this paper, contrast, energy, homogeneity and entropy were calculated which are defined as follows.

The contrast which is a measure of the intensity difference between a pixel and its neighbors over the whole image can be calculated from Eq. (1). Note that the contrast is 0 for a constant image.

$$contrast = \sum_i \sum_j |i - j|^2 GLCM(i, j) \quad (1)$$

Energy is another parameter which is the sum of squared elements in the GLCM.

$$Energy = \sum_i \sum_j GLCM^2(i, j) \quad (2)$$

Homogeneity measures the diagonality of the GLCM and is a measure of smoothness of the texture. It is defined as follows.

$$Homogeneity = \sum_i \sum_j \frac{GLCM(i, j)}{1 + |i - j|} \quad (3)$$

Entropy is a statistical measure of randomness that can be also used to characterize textured regions. It is calculated by:

$$Entropy = - \sum_i \sum_j GLCM(i, j) \log (GLCM(i, j)) \quad (4)$$

These four measures are the inputs to a classifier which determines the class of each pixel (e.g. cement paste, gravel or crack).

2.3 3FLD Algorithm

Fisher's Linear Discriminate Analysis (FLD) is a supervised algorithm used for reducing data dimension before performing a classification. In the supervised methods, the training data have class labels, i.e. data objects that have the form (x_i, y_i) where x_i is the data and y_i shows the class that this data belongs to, for example in a binary classification case $y_i \in \{1, 2\}$. In situations where class labels are available we are often interested in discovering a transformation that maps the data so that different classes are as separated from each other as possible. FLD algorithm maximizes the between class separation and minimizes within class separation.

To perform FLD algorithm, scatter matrices S_i ($i = 1, 2$) and S_w are defined as:

$$S_i = \sum_{x \in \mathcal{D}_i} (x - m_i)(x - m_i)^t \quad (5)$$

Here t stands for transposing a matrix, and

$$S_w = S_1 + S_2 \quad (6)$$

where:

$$m_i = \frac{1}{n_i} \sum_{x \in \mathcal{D}_i} x \quad (7)$$

Between class scatter matrix is also defined by:

$$S_B = (m_1 - m_2)(m_1 - m_2)^t \quad (8)$$

The goal here is to minimize within class scatter and to maximize between class scatter so that we have the best between class separation. Now if the projection matrix is defined by:

$$W = S_w^{-1}(m_1 - m_2) \quad (9)$$

It can be shown that if $y = W^t x$ the within class scatter will be minimized and between class scatter will be maximized. First the transformation matrix is derived from training data then applied to all of the image pixels. Each of classifiers used for comparison will be applied on both FLD algorithm outputs and the original data.

2.4 Classification (Duda, Hart and Stork 2012)

Four different types of classifiers were used in this study. Each of them was applied to both FLD algorithm outputs and original data.

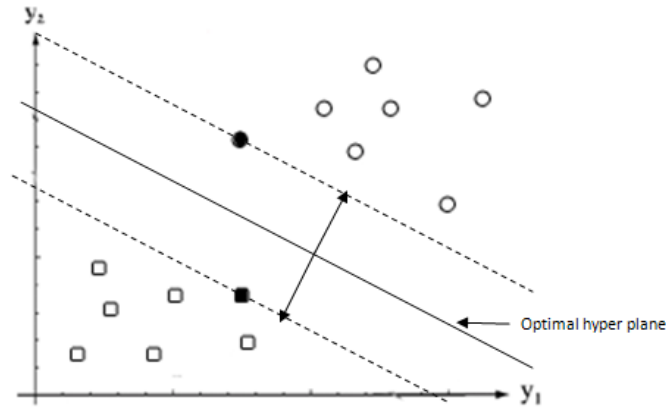


Fig. 3 Finding the optimal hyperplane. Support vectors are nearest patterns. The three support vectors are shown in solid

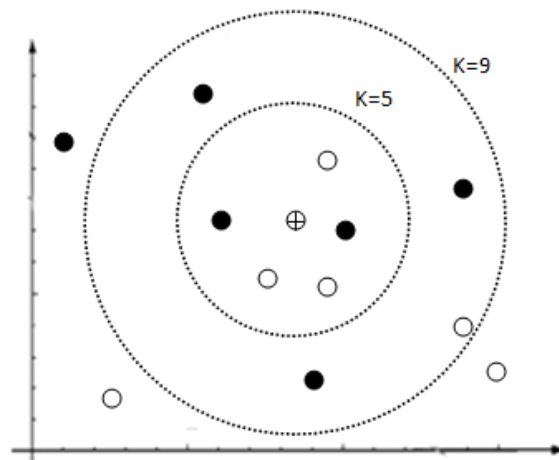


Fig. 4 Effect of K on KNN output

2.4.1 Bayesian classifier

In Bayesian classification, it is assumed that the dataset has a normal probability density. In this case the discriminate functions are defined by:

$$g_i(x) = x^t W_i x + w_i x + w_{i0} \quad i = 1, 2 \quad (10)$$

where

$$W_i = -\frac{1}{2} \Sigma_i^{-1} \quad (11)$$

$$w_i = \Sigma_i^{-1} \mu_i \quad (12)$$

$$w_{i0} = -\frac{1}{2} \mu_i^t \Sigma_i^{-1} \mu_i - \frac{1}{2} \ln |\Sigma_i| + \ln P(\omega_i) \quad (13)$$

$$\mu_i = \frac{1}{n_i} \sum_{x \in \mathcal{D}_i} x \quad (14)$$

$$\Sigma_i = \sum_{x \in \mathcal{D}_i} (x - \mu_i)(x - \mu_i)^t \quad (15)$$

For each pixel x , if $g_1(x) > g_2(x)$ then $x \in \mathcal{D}_1$, otherwise $x \in \mathcal{D}_2$.

2.4.2 Support vector machines

Data from two categories can always be separated by a hyperplane, using a nonlinear transform that maps data to a sufficiently high dimension space. Once the data have been mapped to higher dimension space, there are infinite hyperplanes that can separate the data in two classes, but the question is which one of these hyperplanes is the optimum answer. It can be shown that the plane which has maximum distance from training points of different classes is the best option. Hereby, support vectors are defined as the samples of each class which are closest to separating hyperplane. The decision hyperplane must be equally close to support vectors. Training a Support Vector Machine (SVM) consists of finding the optimal hyperplane, namely the one with maximum distance from the nearest training points as shown in Fig. 3. Although different kernels can be used for SVM training, here a polynomial kernel of degree 3 has been used.

2.4.3 K-Nearest neighbors

In this method, K nearest training points to a test sample are determined. Then the sample is assigned to the class to which majority of these training points belongs. Number of points (K) and the distance function can affect the performance of the algorithm and must be chosen wisely. In the example of Fig. 4, if $k = 5$ the test point (shown with a cross) will be classified as class 1 (white) but if $k = 9$ it will be classified as class 2 (black).

Different distance functions such as Euclidean and Cityblock can be used. In our work, the best results were obtained with $K = 7$ and Euclidean distance.

2.4.4 Artificial neural networks

Artificial neural networks (ANNs) are powerful tools in the pattern recognition. They can be used to model many nonlinear functions. ANNs have been widely used in classification tasks.

A three layer feed-forward network with different number of neurons and transfer functions was tested. Best result was achieved with 50 neurons, tansig (Hyperbolic tangent sigmoid, see Fig. 5) transfer function for hidden layer and logsig (Log-sigmoid, see Fig. 6) function for output layer. Fig. 7 shows an overview of network structure. Levenberg-Marquardt back propagation method with 150 iterations was used for training the network.

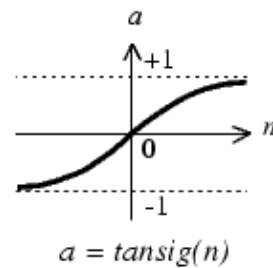


Fig. 5 Hyperbolic tangent sigmoid transfer function

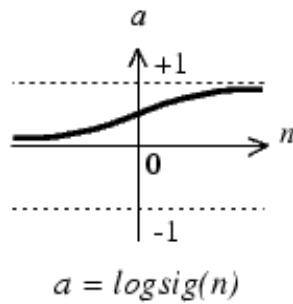


Fig. 6 Log-sigmoid transfer function

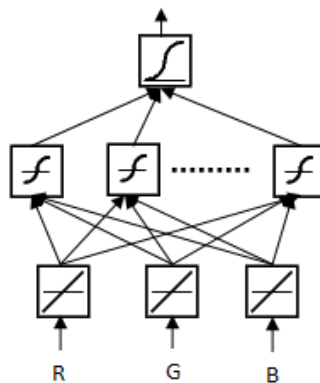


Fig. 7 Overview of network structure

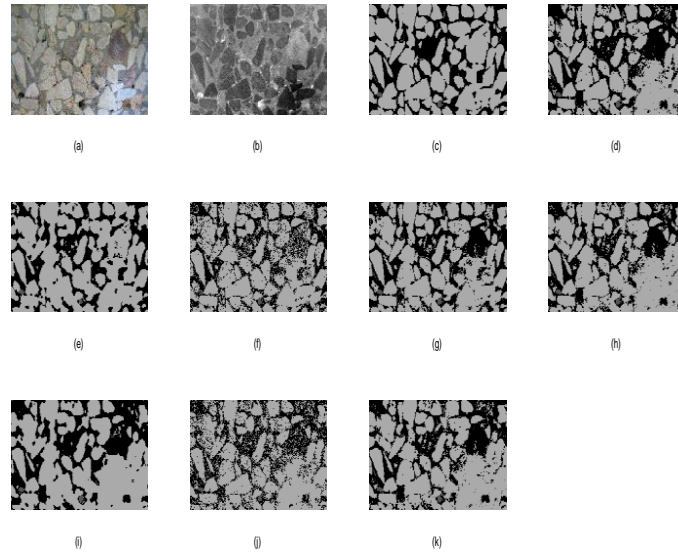


Fig. 8 Segmentation results of different methods a:original image b:FLD output c:hand-segmented image d:segmentation results of Bayesian classifier e: segmentation results of KNN classifier f: segmentation results of neural networks g: segmentation results of SVM image h:segmentation results for combination of Bayesian classifier and FLD i: segmentation results for combination of KNN and FLD j: segmentation results for combination of ANN and FLD k: segmentation results for combination of SVM and FLD

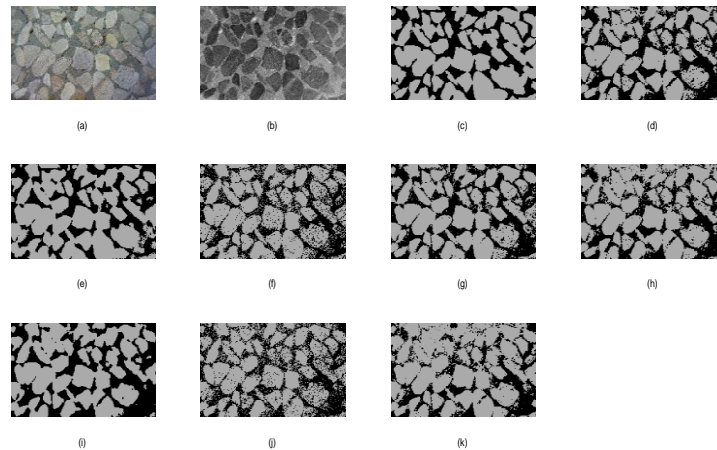


Fig. 9 Segmentation results of different methods a:original image b:FLD output c:hand-segmented image d:segmentation results of Bayesian classifier e: segmentation results of KNN classifier f: segmentation results of neural networks g: segmentation results of SVM image h:segmentation results for combination of Bayesian classifier and FLD i: segmentation results for combination of KNN and FLD j: segmentation results for combination of ANN and FLD k: segmentation results for combination of SVM and FLD

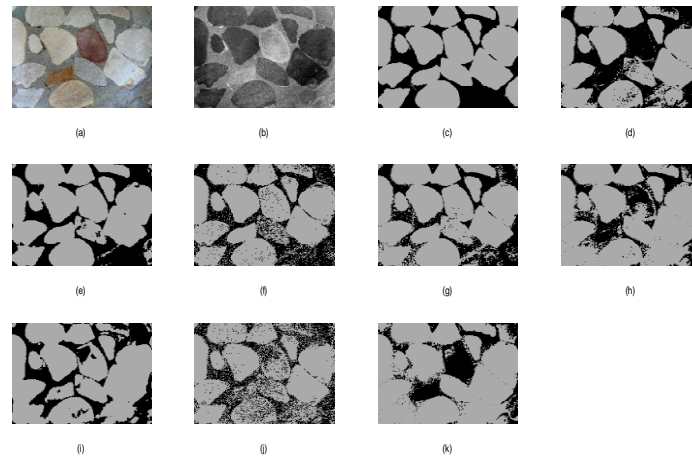


Fig. 10 Segmentation results of different methods a:original image b:FLD output c:hand-segmented image d:segmentation results of Bayesian classifier e: segmentation results of KNN classifier f: segmentation results of neural networks g: segmentation results of SVM image h:segmentation results for combination of Bayesian classifier and FLD i: segmentation results for combination of KNN and FLD j: segmentation results for combination of ANN and FLD k: segmentation results for combination of SVM and FLD

Table 1 Segmentation accuracy for different methods

	Percentage of correct classified pixels	Percentage of gravel pixels classified as cement	Percentage of cement pixels classified as gravel
Bayesian	78.64	-	-
Bayesian + FLD	88.41	7.67	18.04
KNN	88.04	8.87	16.24
KNN+FLD(k=7)	88.67	7.90	16.32
SVM	89.70	6.00	16.89
SVM+FLD	88.58	7.92	16.93
ANN	90.29	5.05	18.69
ANN+FLD	83.84	5.91	33.76

3. Experimental results

For evaluating the performance the methods mentioned in Section 2, sample pictures from concrete cuts were used. All images were segmented using different classification methods once after applying FLD algorithm and once without using FLD.

TABLE I shows segmentation accuracy for each of these methods. Classification accuracy is

the ratio of number of pixels that have been correctly classified to all pixels. ANN is the best method with %90.29 accuracy. KNN and SVM methods also achieve a close accuracy.

As can be seen, FLD has improved the accuracy for Bayesian and KNN classifiers, but it decreases ANN and SVM's performance. This is due to the nonlinear nature of ANN and SVM which causes linear separation of data (like it is done in FLD) to be useless.

Figs. 8 through 10 show segmentation results for some of the input images.

4. Conclusions

Different classifiers were used for the segmentation of concrete images into different phases and their performances were compared. As expected the best results belonged to ANN and SVM.

As seen in TABLE I segmentation accuracy is high and segmented images can be used for further analysis.

Using other kernels for SVM, different structures for ANN and other texture analysis methods is suggested for future researches.

Figure 8. Segmentation results of different methods a:original image b:FLD output c:hand-segmented image d:segmentation results of Bayesian classifier e: segmentation results of KNN classifier f: segmentation results of neural networks g: segmentation results of SVM image h:segmentation results for combination of Bayesian classifier and FLD i: segmentation results for combination of KNN and FLD j: segmentation results for combination of ANN and FLD k: segmentation results for combination of SVM and FLD

Figure 9. Segmentation results of different methods a:original image b:FLD output c:hand-segmented image d:segmentation results of Bayesian classifier e: segmentation results of KNN classifier f: segmentation results of neural networks g: segmentation results of SVM image h:segmentation results for combination of Bayesian classifier and FLD i: segmentation results for combination of KNN and FLD j: segmentation results for combination of ANN and FLD k: segmentation results for combination of SVM and FLD

Figure 10. Segmentation results of different methods a:original image b:FLD output c:hand-segmented image d:segmentation results of Bayesian classifier e: segmentation results of KNN classifier f: segmentation results of neural networks g: segmentation results of SVM image h:segmentation results for combination of Bayesian classifier and FLD i: segmentation results for combination of KNN and FLD j: segmentation results for combination of ANN and FLD k: segmentation results for combination of SVM and FLD

References

- Ammouche, A., Riss, J., Breyse, D. and Marchand, J. (2001), "Image analysis for the automated study of microcracks in concrete", *Cement Concrete Compos.*, **23**(2), 267-278.
- Bernstone, C. and Heyden, A. (2009), "Image analysis for monitoring of crack growth in hydropower concrete structures", *Measurement*, **42**(6), 878-893.
- Carloni, C. and Subramaniam, K.V. (2013), "Investigation of sub-critical fatigue crack growth in FRP/concrete cohesive interface using digital image analysis", *Compos. Part B: Eng.*, **51**, 35-43.
- Chen, Z., Derakhshani, R., Halmen, C. and Kevern, J.T. (2011), "A texture-based method for classifying cracked concrete surfaces from digital images using neural networks", Paper presented at the Neural Networks (IJCNN), The 2011 International Joint Conference on.

- Dequiedt, A.S., Coster, M., Chermant, L. and Chermant, J.L. (2001), "Study of phase dispersion in concrete by image analysis", *Cement Concrete Compos.*, **23**(2), 215-226.
- Dequiedt, A.-S., Redon, C., Chermant, J.L., Chermant, L. and Coster, M. (2013), Characterisation of diffusion paths of water in concrete by color image analysis, *Acta Stereologica*, 18(2).
- Duda, R.O., Hart, P. E. and Stork, D.G. (2012), *Pattern classification*: John Wiley & Sons.
- Feng, X., Garboczi, E., Bentz, D., Stutzman, P. and Mason, T.O. (2004), "Estimation of the degree of hydration of blended cement pastes by a scanning electron microscope point-counting procedure", *Cement Concrete Res.*, **34**(10), 1787-1793.
- Glinicki, M., & Litorowicz, A. (2006), "Crack system evaluation in concrete elements at mesoscale", *Tech. Sci.*, 54(4).
- Haralick, R.M., Shanmugam, K. and Dinstein, I.H. (1973), "Textural features for image classification", *Systems, Man and Cybernetics*, IEEE Transactions on, (6), 610-621.
- Liang, Z., Changhua, L., Dengfeng, C. and Faning, D. (2009), *Concrete CT Image Segmentation Using Modified Metropolis Dynamics*, Paper presented at the Image and Signal Processing, 2009. CISP'09. 2nd International Congress on.
- Litorowicz, A. (2006), "Identification and quantification of cracks in concrete by optical fluorescent microscopy", *Cement Concrete Res.*, **36**(8), 1508-1515.
- Martins, A., Pizolato Junior, J. and Belini, V. (2013), "Image-based method for monitoring of crack opening on masonry and concrete using mobile platform. Revista IBRACON de estruturas e materiais", **6**(3), 414-435.
- Petrou, M., Sevilla, P.G. and Wiley, J. (2006), *Image processing: dealing with texture* (Vol. 10): Wiley Chichester.
- Redon, C., Chermant, L., Chermant, J.L. and Coster, M. (1999), "Automatic image analysis and morphology of fibre reinforced concrete", *Cement Concrete Compos.*, **21**(5), 403-412.
- Ringot, E. and Bascoul, A. (2001), "About the analysis of microcracking in concrete", *Cement Concrete Compos.*, **23**(2), 261-266.
- Sarafrazi, K., Yazdi, M. and Abedini, M.J. (2013), "A new image texture segmentation based on contourlet fractal features", *Arabian J. Sci. Eng.*, **38**(12), 3437-3449.
- Shamshad, A., Johari, M.M., Hussin, W.W., Sanusi, S.M. and Majid, T. (2013), "Digital image analysis of cracks in concrete of different grades", *Flood Risk Management: Research and Practice: Extended Abstracts Volume* (332 pages)+ full paper CD-ROM, 212.
- Silva, A.G., de Alencar Lotufo, R. and Flores, F.C. (2002), "Classification of microstructures by morphological analysis and estimation of the hydration degree of cement paste in concrete", *Paper presented at the Computer Graphics and Image Processing, 2002. Proceedings*, XV Brazilian Symposium on.
- Soroshian, P. and Elzafraney, M. (2005), "Morphological operations, planar mathematical formulations, and stereological interpretations for automated image analysis of concrete microstructure", *Cement Concrete Compos.*, **27**(7), 823-833.
- Soroshian, P., Elzafraney, M. and Nossoni, A. (2003), "Specimen preparation and image processing and analysis techniques for automated quantification of concrete microcracks and voids", *Cement Concrete Res.*, **33**(12), 1949-1962.
- Tuceryan, M. and Jain, A.K. (1993), *Texture analysis. Handbook of pattern recognition and computer vision*, 276.
- Wong, H., Head, M. and Buenfeld, N. (2006), "Pore segmentation of cement-based materials from backscattered electron images", *Cement Concrete Res.*, **36**(6), 1083-1090.
- Yang, R. and Buenfeld, N. (2001), "Binary segmentation of aggregate in SEM image analysis of concrete", *Cement Concrete Res.*, **31**(3), 437-441.