Journal of Hazardous Materials 223–224 (2012) 94–103

Contents lists available at SciVerse ScienceDirect

Journal of Hazardous Materials

journal homepage: www.elsevier.com/locate/jhazmat

Particulate matter characterization by gray level co-occurrence matrix based support vector machines

K. Manivannan [a], P. Aggarwal [a], V. Devabhaktuni [a,*], A. Kumar [b], D. Nims [b], P. Bhattacharya [c]

[a] EECS Department, University of Toledo, MS 308, 2801 W Bancroft Street, Toledo, OH 43606, USA
[b] Department of Civil Engineering, University of Toledo, MS 307, 2801 W Bancroft Street, Toledo, OH 43606, USA
[c] School of Computing Sciences & Informatics, University of Cincinnati, 814B Rhodes Hall, Cincinnati, OH 45221, USA

ARTICLE INFO

Article history:
Received 1 December 2011
Received in revised form 11 April 2012
Accepted 24 April 2012
Available online 30 April 2012

Keywords:
Particulate matter
Support vector machines
Gray level co-occurrence matrix
Image segmentation

ABSTRACT

An efficient and highly reliable automatic selection of optimal segmentation algorithm for characterizing particulate matter is presented in this paper. Support vector machines (SVMs) are used as a new self-regulating classifier trained by gray level co-occurrence matrix (GLCM) of the image. This matrix is calculated at various angles and the texture features are evaluated for classifying the images. Results show that the performance of GLCM-based SVMs is drastically improved over the previous histogram-based SVMs. Our proposed GLCM-based approach of training SVM predicts a robust and more accurate segmentation algorithm than the standard histogram technique, as additional information based on the spatial relationship between pixels is incorporated for image classification. Further, the GLCM-based SVM classifiers were more accurate and required less training data when compared to the artificial neural network (ANN) classifiers.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Particulate matter (PM) is highly complex mixture of very fine particles and liquid droplets present in the atmosphere. It is one of the six criteria air pollutants identified and monitored by the U.S. Environmental Protection Agency (EPA) [1]. PM can be classified as ultrafine, fine, or thoracic particles depending on their size. These particles exist in different shapes and densities in the air, and hence, the aerodynamic diameter has been recognized as a simple means of defining the size of particles [1,2]. The particles can penetrate deep into the lungs and cause degradation of lung function, breathing problems, irregular heartbeat and nonfatal heart attacks. Hence, it is vital to determine the number, morphology and size distribution of these particles so as to reflect their intensity of harmful effects on human health.

Several techniques are developed to analyze the physical and chemical properties of PMs such as gravimetric method [3], atomic absorption spectroscopy [4], and high performance liquid, gas chromatography [5]. Scanning electron microscope (SEM), computer controlled scanning electron microscope and transmission electron microscope have been used to provide quantitative information on the size, morphology and elemental composition of particles [6].

The SEM method produces magnified images, better feature resolution and greater depth of field as compared to light microscope [7]. Various commercial software tools for analyzing the SEM-based particle images are available in the market [8]. The main limitation of these packages is their inability to automatically determine the best segmentation algorithm for a given image.

Segmentation refers to the process of partitioning a digital image into multiple segments so as to separate the particles from the background in the image [9]. For each image, the segmentation algorithm is manually selected from a stack of available algorithms based on user's experience and knowledge. After segmenting the image, particle features can be easily extracted.

Feature extraction is the process of generating features from the image for classification and can be implemented by various techniques such as histogram or gray level co-occurrence matrix (GLCM). Histogram based methods are widely used for feature extraction since they are fast and simple methods. Features derived from this approach include moments such as mean, standard deviation, average energy, etc. [10]. However, in the histogram approach the relationship with the neighboring pixels is not considered, which limits its performance. An alternate method is the GLCM method, which determines the spatial relationship between the pixels by calculating the difference in intensities between the

* Corresponding author. Tel.: +1 419 530 8172; fax: +1 419 530 8146.
E-mail addresses: ktiruma@rockets.utoledo.edu (K. Manivannan), Priyanka.Aggarwal@utoledo.edu (P. Aggarwal), Vijay.Devabhaktuni@utoledo.edu (V. Devabhaktuni), Ashok.Kumar@utoledo.edu (A. Kumar), Douglas.Nims@utoledo.edu (D. Nims), bhattapr@ucmail.uc.edu (P. Bhattacharya).

0304-3894/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.jhazmat.2012.04.056

center pixel and its neighbors [11]. Other available methods include gray level run-length method, gray level difference matrix and Power spectrum method [12]. The GLCM-based methods outperformed the other conventional segmentation methods in various applications such as terrain classification, lamb grading and medical image analysis [13–15], mainly due to utilization of the second order probabilities.

In this paper, a GLCM-based support vector machine (SVM) method is proposed which is an improved version of our previous work using the histogram-based SVM method [8]. The GLCM-based technique quickly learns the additional knowledge obtained using spatial relations in an image and applies it for selecting the best segmentation algorithm. There are a number of machine learning techniques for image classification like artificial neural networks (ANN), fuzzy logic, genetic algorithm, and SVMs. The structure of ANNs is difficult to understand and they may fail to capture unique attributes in the training phase [16]. Fuzzy logic requires prior knowledge about the system, while genetic algorithms suffer complications in adequately representing the training/output data [24]. Out of various available machine learning techniques, SVM is chosen because of its high generalization capability along with no additional knowledge requirement, even with high dimensionality of the input space [17,18].

In [8], the SVM is trained using the intensity histogram of the image (upper panel in Fig. 1), which carries no information regarding the relative position of the pixels with respect to each other. Relative position of pixels is important for describing image features and one way to incorporate this is through GLCM method (lower panel in Fig. 1). In this paper, the proposed GLCM-based SVM approach illustrates higher accuracy than the histogram-based method for automating the image segmentation process.

The rest of the paper is organized as follows. The process of segmentation and feature extraction are explained in Section 2. Section 3 briefly describes the concept of SVMs while the ANN model is discussed in Section 4. The results and discussion is given in Section 5 while Section 6 concludes the paper.

2. Particulate matter characterization methodology based on SEM

Generally, SEM images of the filter paper (on which particles are collected) are represented by a 2D array of pixels followed by noise reduction, segmentation and feature extraction procedures. Out of these processes, segmentation is the most vital step in analyzing images for PM characterization.

2.1. Segmentation

Segmentation is the process of converting a digital image into semantically interpretable regions by locating the boundaries such as lines and curves of the visible particles [19]. Segmentation divides an image into its constituent regions or objects such that a set of connected pixels have similar properties like color, intensity or texture. In this process, labels are assigned to each image pixel and the pixels with same label are grouped as either particle pixel or background pixel. Various segmentation techniques usually fall under four main categories, namely, clustering, thresholding, edge-detection and region-based methods [20].

The clustering method groups pixels into clusters and distinguishes them by color or texture. The thresholding method converts multilevel images into binary images (black and white). The edge detection method locates edges of the object in the image while the region-based technique distinguishes the individual pixel from the group of pixels [20]. In this paper, the thresholding methods like Kapur [21], Rosin [22], Otsu [23], Minimum error [24] and one edge detection method known as Sobel [25] are incorporated to evaluate the best segmentation algorithm for images. However, one single segmentation algorithm may not produce the best result for all types of images. Hence, there is a need for an automatic selection process of the optimal segmentation algorithm for a given image to reduce human intervention.

2.2. Feature extraction and textural representation

Feature extraction is the measurement of a quantifiable property, which specifies significant characteristics of the object [26]. The extracted features are represented as vectors and have a high influence on the classification efficacy. An important approach to obtain feature description is the object texture content measurement, which is further divided into structural and statistical categories. Structural techniques deal with the arrangement of image primitives whereas the statistical method yields characteristics of the image texture. The statistical method is a simple approach, which analyzes the spatial distribution of the image gray values by determining the local features of each image and derives a set of statistics (first-order, second-order or higher-orders) from the pixel distribution [27]. The first-order method evaluates the properties of the individual pixels only, while second-order and higher-order methods estimate the pixel properties occurring at different locations relative to each other. The histogram method belongs to the first-order approach while the GLCM method follows the second-order approach.

2.2.1. Histogram

A histogram is a graphical representation containing the tonal distribution of all image pixels. The histogram of a digital image with intensity levels in the range of [0,L–1] is a discrete function with L being the number of gray levels and is given by

$$h(r_k) = n_k, \tag{1}$$

where r_k is the kth intensity gray level value and n_k is the number of pixels in the image with intensity r_k. In general, the intensity levels are represented as 8-bit integers in the range 0–255 [28]. A histogram is made up of bins carrying certain intensity values in each bin. The pixels with identical intensity values are grouped together and the total number of pixels lying in each intensity range is assigned to the respective bins. However, the drawback of the histogram in classification of images is that the method is totally dependent on the intensity value and number of pixels, ignoring the content or shape information. The histogram for two different images can potentially be identical, since they may share the same grayscale information, but may have totally different object content.

2.2.2. Gray level co-occurrence matrices

GLCM is the method of computing the frequency of pixel pairs having the same gray level in the image [28]. The relationship between the reference pixel and the neighboring pixels is calculated to determine the textural features of the image. In Fig. 1 (lower panel), the cells 1 and 5 are the nearest neighbors at 0° to cell 0. Similarly, cells 2 and 6, 3 and 7, 4 and 8 are the nearest neighbors at 135°, 90° and 45°, respectively, to cell 0. The GLCM is formed by evaluating the count of pixel pairs with gray level value i occurring adjacent to the pixel with gray level value j [29]. The relative frequencies of gray level pixel pairs separated by a distance d in a particular angle θ forms the displacement vector (d, θ).

Let $f: V_y \times H_x \rightarrow I$ be an image with dimensions, $V_y = (0, 1, 2, \ldots, I_y - 1)$, $H_x = (0, 1, 2, \ldots, I_x - 1)$ having a set of quantized gray-tones $G = (0, 1, 2, \ldots, L - 1)$. The coordinates of this image $V_y \times H_x$ represent the resolution cells containing the gray values for each pixel. The texture is assessed by the four closely related measures called

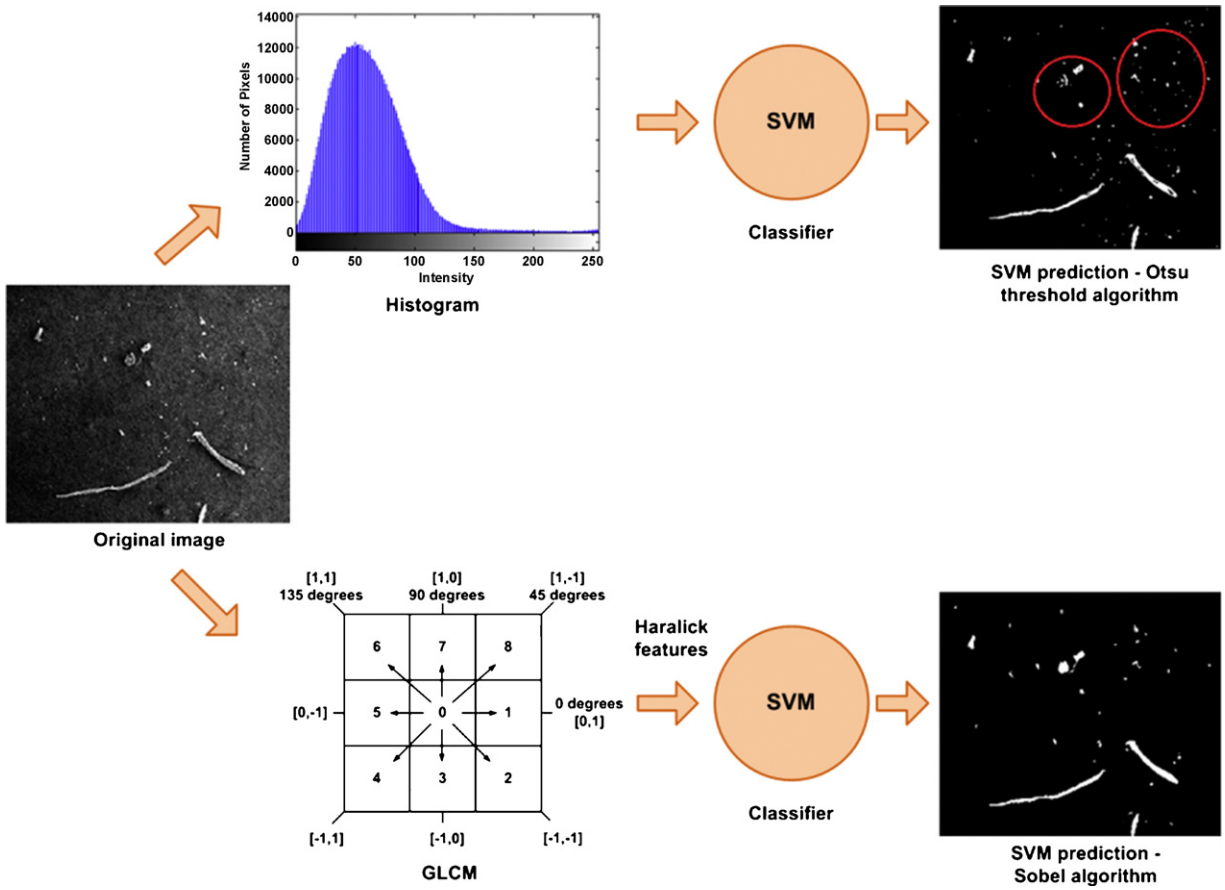

Fig. 1. Image classification using histogram-based SVM and GLCM-based SVM.

angular nearest neighbor gray tone spatial dependence matrices. The content information between two neighboring cells $\{(k, l), (m, n)\}$ separated by a distance d is represented by C_{ij} as illustrated in Fig. 2. Since the textures involved in the images considered in this study are micro-textures, the distance $d = 1$ is chosen in order to extract the detailed textural information. Moreover, the pixels are more likely to correlate with the one that is located closer (i.e., with smaller value of d) than the pixels that are situated far away [30]. In the figure, the displacement vector is (1,0°) with $k = m = 1$ and $l = 1$, $n = 2$ respectively, where $I(k, l) = i$, $I(m, n) = j$. Thus the gray tone spatial dependence matrix is the function of the angular relationship between the neighboring resolution cells and the distance between them.

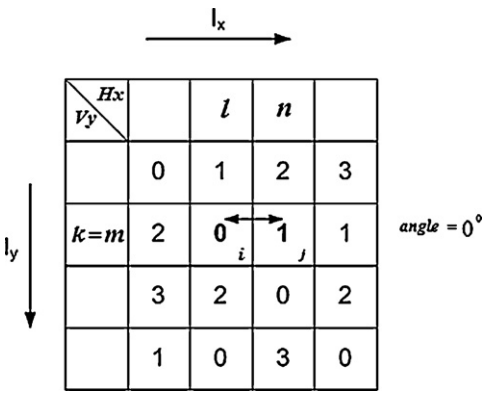

Fig. 2. Example of an image gray-tone spatial dependence matrix with levels 0–3.

The spatial measurement at different angles with the distance d is given in Eqs. (2)–(5).

$$C(i, j, d, 0^\circ) = \#\{(k, l), (m, n) : k - m = 0, \left|l - n\right| = d\} \tag{2}$$

$$\begin{gathered} C(i, j, d, 45^\circ) = \#\{(k, l), (m, n) : k - m = d, l - n = -d\} \\ \text{or} \\ C(i, j, d, 45^\circ) = \#\{(k, l), (m, n) : k - m = -d, l - n = d\} \end{gathered} \tag{3}$$

$$C(i, j, d, 90^\circ) = \#\{(k, l), (m, n) : |k - m| = d, l - n = 0\} \tag{4}$$

$$\begin{gathered} C(i, j, d, 135^\circ) = \#\{(k, l), (m, n) : k - m = d, l - n = d\} \\ \text{or} \\ C(i, j, d, 135^\circ) = \#\{(k, l), (m, n) : k - m = -d, l - n = -d\} \end{gathered} \tag{5}$$

where # represents the count of pixel pairs with identical intensity level.

In Fig. 3(a), the matrix element present in the first row second column (i.e., 3) represents the total number of instances that gray tones of value $i = 0$ and $j = 1$ occurred horizontally adjacent to each other as in Fig. 2. Here, the entry at the coordinates (0,1) is 3 because there are three pixel pairs of (0,1) with distance $d = 1$ at angle 0°. Similarly, the spatial dependence matrix at angles 90°, 135° and 45° are shown in Fig. 3(b–d).

Further, the spatial dependence of gray levels is quantified by calculating the fourteen textural features of the co-occurrence matrix constructed by Haralick et al. [11]. These fourteen textural parameters are highly correlated and hence only four features in Eqs. (6)–(9) are sufficient for obtaining good image classifications [13,31].

$$\text{a}\begin{pmatrix}0&3&3&2\\3&2&1&0\\3&1&0&2\\2&0&2&0\end{pmatrix}\quad \text{b}\begin{pmatrix}0&2&4&1\\2&0&2&2\\4&2&0&1\\1&2&1&0\end{pmatrix}$$

$$\text{c}\begin{pmatrix}6&0&0&1\\0&2&2&0\\0&2&2&1\\1&0&1&0\end{pmatrix}\quad \text{d}\begin{pmatrix}2&1&1&1\\1&0&3&1\\1&3&0&1\\1&1&1&0\end{pmatrix}$$

Fig. 3. Co-occurrence matrices derived from the spatial dependence matrix shown in Fig. 2 for angles: (a) 0°, (b) 90°, (c) 135°, (d) 45°.

Energy: The energy measures the occurrence of repeated pixel pairs within the image.

$$\text{Energy} = \sum_i \sum_j C(i,j)^2 \tag{6}$$

Contrast: The contrast measures the difference between the maximum and minimum value of a contiguous set of pixels.

$$\text{Contrast} = \sum_i \sum_j (i-j)^2 C(i,j) \tag{7}$$

Entropy: The entropy measures the disorder of the image and is highly correlated to energy. Non-uniform texture has a high entropy value.

$$\text{Entropy} = -\sum_i \sum_j C(i,j)\log(C(i,j)) \tag{8}$$

Inverse difference: The inverse difference (ID) measures the smoothness of the image.

$$\text{ID} = \sum_i \sum_j \frac{C(i,j)}{1+(i-j)^2} \tag{9}$$

Once these parameters are obtained for each image, SVMs are trained and then tested for evaluating the best segmentation algorithm with randomly selected images.

3. Support vector machines

SVMs are supervised learning algorithms developed by Cortes and Vapnik [32]. Given a set of training data belonging to two different categories, an SVM training algorithm builds a training model that predicts the class for the new data. SVMs work on statistical learning theory and can produce robust, accurate and effective results using small training samples. The SVM determines the maximum-margin hyperplane to separate two classes based on the segmentation algorithm (Fig. 4). The decision hyperplane consists of an intercept term b and a normal vector perpendicular to the hyperplane called the weight vector $\boldsymbol{w}$ and is given by

$$\boldsymbol{w}.y + b = 0 \tag{10}$$

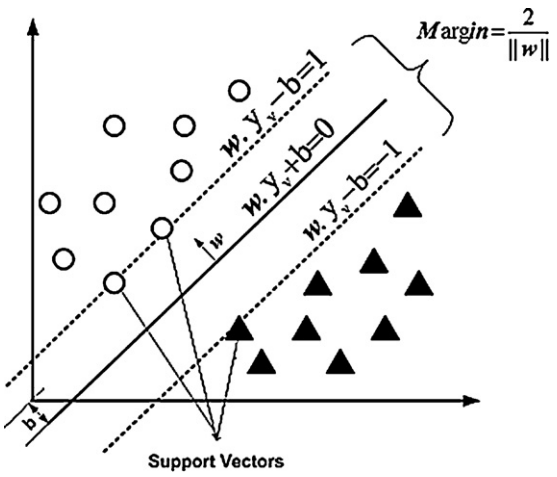

Fig. 4. Schematic of a hyperplane seperating two classes.

where y is any random set of data points. Hence, to maximize the margin between hyperplanes while still separating the data, optimal $\boldsymbol{w}$ and b parameters are determined by

$$\boldsymbol{w} = \sum_{v=1}^{u} \alpha_v x_v y_v, \tag{11}$$

and

$$b = \frac{1}{N_s}\sum_{v=1}^{N_s}(\boldsymbol{w}.y_v - x_v) \tag{12}$$

where α is non-negative Lagrange multiplier, $x_v \in \{-1, +1\}$ is the class label, u is the number of training sets and N_s is the number of support vectors.

SVMs are fundamentally designed for binary class problem, hence for a data set with s number of segmentation algorithms, SVMs assume $s = 1$ as class one and the remaining labels as class two. SVMs then find the maximum margin hyperplane between class one and class two using Eq. (10). The value of s is incremented till it reaches the maximum number of s classes for a multi-class problem. Thus, SVMs are able to create a training model file by assigning the classes for each training image. During the testing phase, this learnt knowledge is applied to classify the images. The classification performance of SVM depends mainly on the selection of kernel function and the regularization parameter C. Out of the different types of kernel functions, the commonly used radial basis function (RBF) was selected because of its faster training rate [33]. Further, to evaluate the optimal values for the kernel function parameters and C, the Leave One Out cross-validation method [34] was utilized that determined the optimal width of RBF to be 25 and C equal to 1000. Below is the detailed methodology for SVM training and testing.

3.1. Training histogram-based and GLCM-based SVMs

The histogram is created by aggregating the pixels having similar intensity values in each of the 256 bins. The training dataset is then prepared containing one "target value" (class label or the segmentation algorithm) and several "attributes" (features) with the following format.

$$D = \{(x^1, y^1_{1x256}), (x^2, y^2_{1x256}), \ldots, (x^s, y^t_{1x256})\}^T \tag{13}$$

where T represents the transpose function, t is the total number of training images, $x^1, x^2, \ldots, x^s$ are the segmentation algorithms and

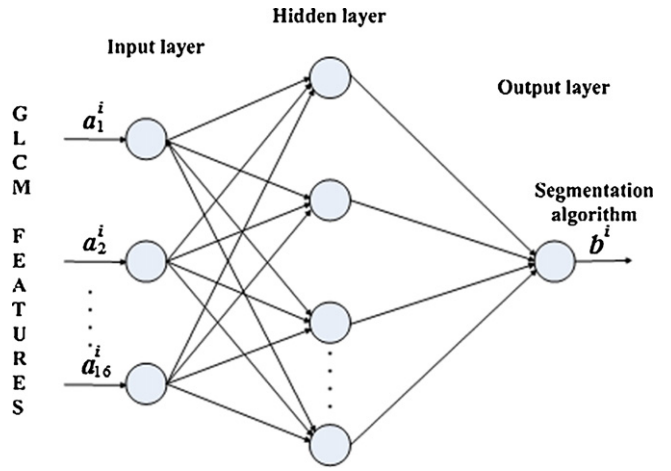

Fig. 5. GLCM-based ANN model using MLP network.

$y^1, y^2, \ldots, y^t$ are feature vectors for t training images each with 256 entries.

On the other hand, the GLCM is calculated for each image at four different angles, namely, 0°, 45°, 90° and 135°. The Haralick parameters (refer to Section 2.2.2) of entropy, energy, contrast and inverse difference are evaluated for these four angles to create a feature vector of 16 bins for an image as given by

$$D = \{(x^1, z^1_{1x16}), (x^2, z^2_{1x16}), \ldots, (x^s, z^t_{1x16})\}^T \quad (14)$$

where $z^1, z^2, \ldots, z^t$ are the feature vectors each containing 16 bins.

3.2. Testing SVM

Once SVM is sufficiently trained, it is tested using a dataset created based on the image intensity histogram and the GLCM matrix. Assuming m testing images, the histogram and the GLCM data are given in Eqs. (15) and (16)

$$D = \{y^1_{1x256}, y^2_{1x256}, \ldots, y^t_{1x256}\}^T \quad (15)$$

$$D = \{z^1_{1x16}, z^2_{1x16}, \ldots, z^t_{1x16}\}^T \quad (16)$$

To compare the performance of the SVM algorithm, an ANN model is also implemented in this paper as described below.

4. Artificial neural network

ANNs are non-linear statistical data model capable of learning the relationships between a series of input and the corresponding outputs. ANNs contain a number of individual units (called neurons) that are inter-connected with their associated weights. The network parameters, like weights and biases, are adjusted through a training process. A common type of ANN is a multi-layer perceptron (MLP) network, consisting of input, hidden and output layers. Experience along with trial and error method helps in choosing the number of hidden neurons in the layer while training data decides the neurons in the input and output layers.

In this study, an MLP network with one hidden layer is trained using back propagation algorithm to derive a model for selecting the optimal image segmentation algorithm (Fig. 5). To derive the input–output functional relationship model (f), the MLP network is initially assigned random weights and is then trained with the desired input and output patterns. The relationship to be modeled between the input $\boldsymbol{a}$ and the output $\boldsymbol{b}$ is given in Eq. (17)

$$b = f(\boldsymbol{a}, \boldsymbol{w}) \quad (17)$$

where $\boldsymbol{w}$ is the parameter which is estimated optimally during training. The main objective of this training is to determine the suitable weight $\boldsymbol{w}$ that minimizes the training error $\{E(w)\}$ given in Eq. (18) [35].

$$E(w) = \frac{1}{2}\sum_{i=1}^{t}[\hat{b}^i - b^i]^2 \quad (18)$$

where $\hat{b}^i$ is the trained output corresponding to the input $a^i_{1\times16}$, i varies from 1 to t (total number of training images) and b^i is the desired output.

4.1. Training and testing ANN model

An ANN model is created such that the training error is minimized and is less than 10^{-2} with the maximum of 200 iterations. The training sets for ANN using GLCM method containing the input feature and the corresponding output segmentation algorithm is shown in Eq. (19) and validating sets created to test ANN is given in Eq. (20).

$$A = \{(a^1_{1x16}, b^1), (a^2_{1x16}, b^2), \ldots, (a^t_{1x16}, b^t)\}^T \quad (19)$$

$$A = \{(a^1_{1x16}), (a^2_{1x16}), \ldots, (a^t_{1x16})\}^T \quad (20)$$

In this study, the number of input neurons is set to 16, the number of hidden neurons to 25 and the output neuron to 1. The number of hidden neurons is determined using the trial and error method varying between 5 and 50 until a minimum training error percentage is obtained. Finally the samples are validated to determine the classification accuracy.

5. Results and discussion

To study the characteristics of the particulate matter, the airborne particles are collected on the polytetrafluoroethylene (PTFE) filter paper using Grimm 1.108 aerosol spectrometer. FEI Quanta 3D SEM then divided the filter paper into several grids and captured the images from each grid [3]. Particles appeared as bright spots in these images. The feasible images are selected from the stack of captured images and a database for randomly selected training and testing samples is prepared. Three cases are implemented and assessed: Case 1 consists of three segmentation algorithms to choose from, namely, Minimum error threshold, Otsu threshold and Rosin threshold. Similarly, Case 2 consists of four algorithms, namely Minimum error threshold, Sobel, Otsu and Kapur algorithms. For Case 3, the samples are segmented by five segmentation algorithms, namely, Kapur threshold, Rosin threshold, Minimum error threshold, Otsu threshold and Sobel algorithms. Equal number of training images is used for each segmentation algorithm for fair assessment of the trained SVM classifier.

5.1. Classification performance of GLCM-based and histogram-based SVMs

A database containing 150 images was selected and randomly divided into training and testing samples. Fig. 6 shows the segmented images after applying three algorithms of Case 1. The SVM predicted the Minimum error threshold as the best segmentation method when trained using GLCM, whereas, Rosin was predicted as best algorithm when trained using histogram method. This resulted into better performance of GLCM-based SVM method compared to the histogram-based SVM method. The best segmentation algorithm is manually selected by the expert by comparing the segmented image with prominent particles and the original image. Note that the image shown in Fig. 6(a) is randomly selected from a

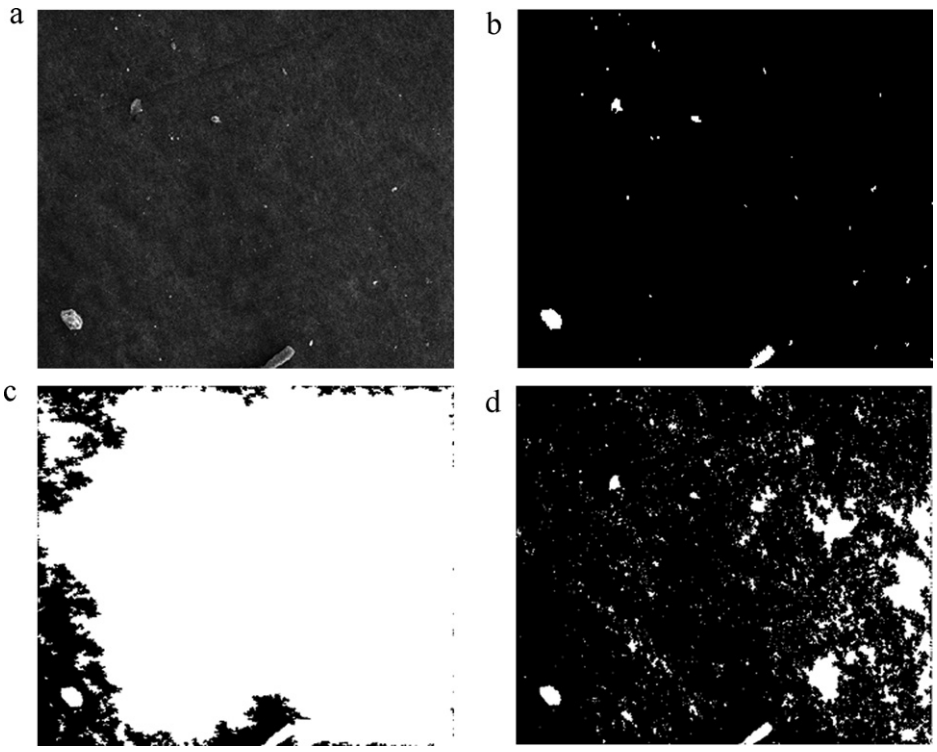

Fig. 6. Performance of segmentation algorithms used in Case 1: (a) original image. (b) Segmentation using Minimum error threshold method predicted by GLCM-based SVM (c) Otsu threshold segmentation, (d) Rosin threshold segmentation predicted by histogram-based SVM.

Table 1
Overall performance of histogram-based and GLCM-based SVM for Case 1.

No. of training images	No. of testing images	No. of correctly predicted images		No. of incorrectly predicted images		Accuracy (%)	
		Histogram	GLCM	Histogram	GLCM	Histogram	GLCM
25	125	47	97	78	28	37	77
50	100	64	79	36	21	64	79
75	75	49	63	26	12	65	84
100	50	34	47	16	3	68	94
125	25	20	24	5	1	80	96

set of 150 images and the segmented images using different algorithms are shown in Fig. 6(b–d). Fig. 6(b) is perceived by the expert to be the best segmented image among others, as the particles present in the image are protuberant compared to the other image shown in Fig. 6(c) and (d). Table 1 depicts the number of images correctly and incorrectly predicted by SVM classifiers. For 25 training and 125 testing images, the GLCM-based method gave 77% accuracy while the histogram-based method gave 37%. Further, on increasing the training images to 125, GLCM-based SVM gave 96% accuracy as opposed to 80% by histogram method. This clearly demonstrates that the number of images correctly predicted by GLCM is more as compared to the histogram method. Further, GLCM-based SVM required only 12–40 iterations for training as opposed to the histogram method which required 120–500 iterations for the same number of samples. For instance, GLCM and histogram needed 12 and 122 iterations respectively for 25 training samples.

For Case 2 (Fig. 7), GLCM-based SVM gave 96% and histogram-based SVM gave 68% accuracy with 125 training and 25 testing images. The segmentation by Sobel method produced much cleaner results for this particular image compared to the other algorithms (red circles) which was accurately predicted by GLCM-based SVM. From Table 2, it is observed that with the increase in the number of training images from 25 to 125, the GLCM method accuracy increased from 76% to 96%.

For Case 3, we incorporated five segmentation algorithms to train histogram-based and GLCM-based SVMs. In this Case (Fig. 8), GLCM-based SVM yielded 88% accuracy compared to 52% in histogram-based SVM with 125 training and 25 test images. From Table 3, we can clearly establish that the GLCM-based method is more accurate than the histogram-based method for all the cases. Tables 1, 2 and 3 also illustrate that the accuracy produced by applying three segmentation algorithms is higher than that obtained by five segmentation algorithms.

Fig. 9 shows a monotonic increase in accuracy of GLCM-based SVM with increase in number of training images. Further, the accuracy decreases when more segmentation algorithms are included in SVM. The accuracy of SVM was higher for three algorithms (Case1) which decreased when the number of algorithms was increased to four (Case 2) and five (Case3).

Table 4 presents the evaluated number of particles present in the image shown in Fig. 7(a) with different segmentation algorithms. The number of particles in the image is determined using the Matlab built-in function called 'bwlabel' which computes connected components in a binary image. For the randomly chosen image shown in Fig. 7(a), the Sobel algorithm was selected as the best algorithm by the expert and was correctly predicted by the GLCM-based SVM. For this image, the Sobel algorithm identified 20 particles which are closest to the actual 31 particles in the original

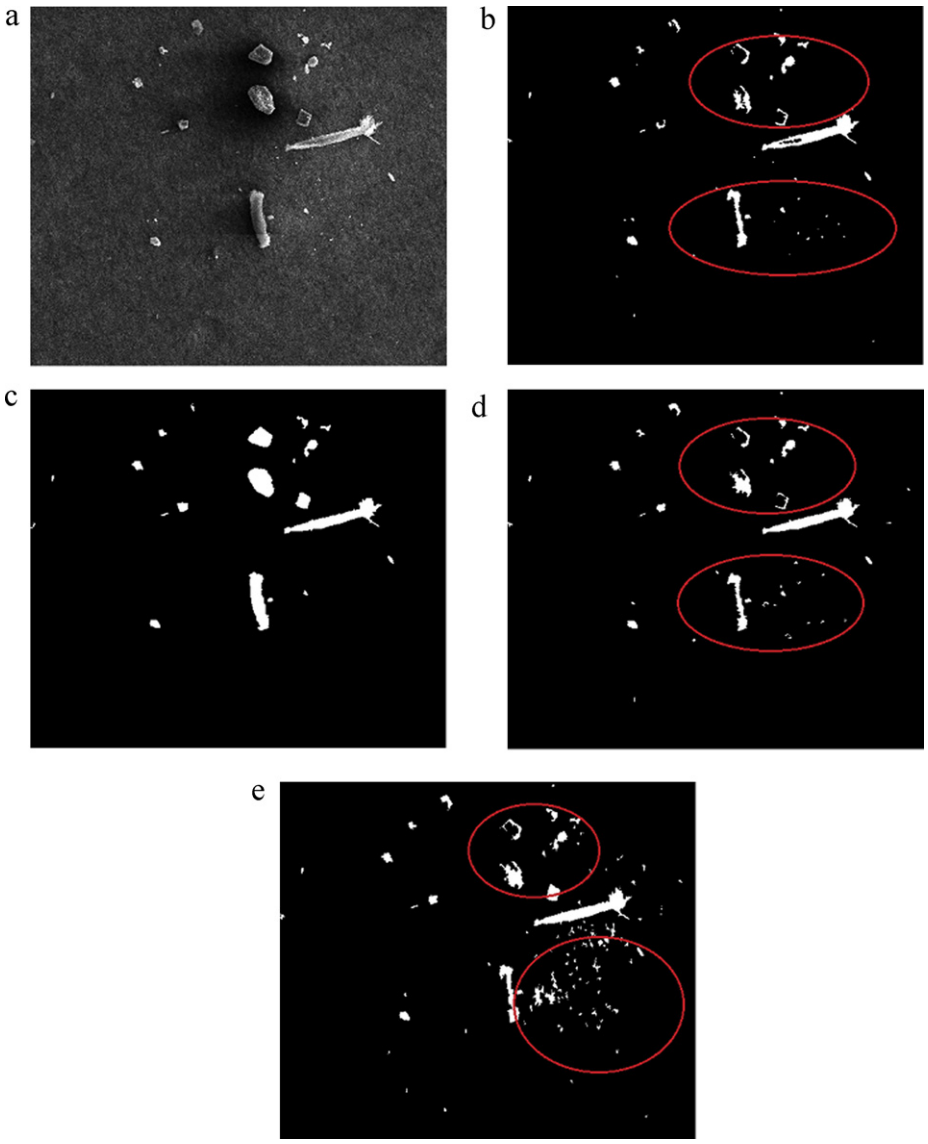

Fig. 7. Performance of segmentation algorithms used in Case 2:(a) original image, (b) Minimum error threshold predicted by histogram-based SVM, (c) Sobel method predicted by GLCM-based SVM, (d) Otsu threshold, (e) Kapur threshold. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

Table 2
Overall performance of histogram-based and GLCM-based SVM for Case 2.

No. of training images	No. of testing images	No. of correctly predicted images		No. of incorrectly predicted images		Accuracy (%)	
		Histogram	GLCM	Histogram	GLCM	Histogram	GLCM
25	125	44	95	81	30	35	76
50	100	60	79	40	22	60	78
75	75	46	63	29	13	62	83
100	50	33	47	17	6	66	88
125	25	25	24	8	1	68	96

Table 3
Overall performance of histogram-based and GLCM-based SVM for Case 3.

No. of training images	No. of testing images	No. of correctly predicted images		No. of incorrectly predicted images		Accuracy (%)	
		Histogram	GLCM	Histogram	GLCM	Histogram	GLCM
25	125	41	94	84	31	33	75
50	100	59	75	41	25	59	75
75	75	45	59	30	16	60	79
100	50	32	42	18	8	64	84
125	25	13	22	12	3	52	88

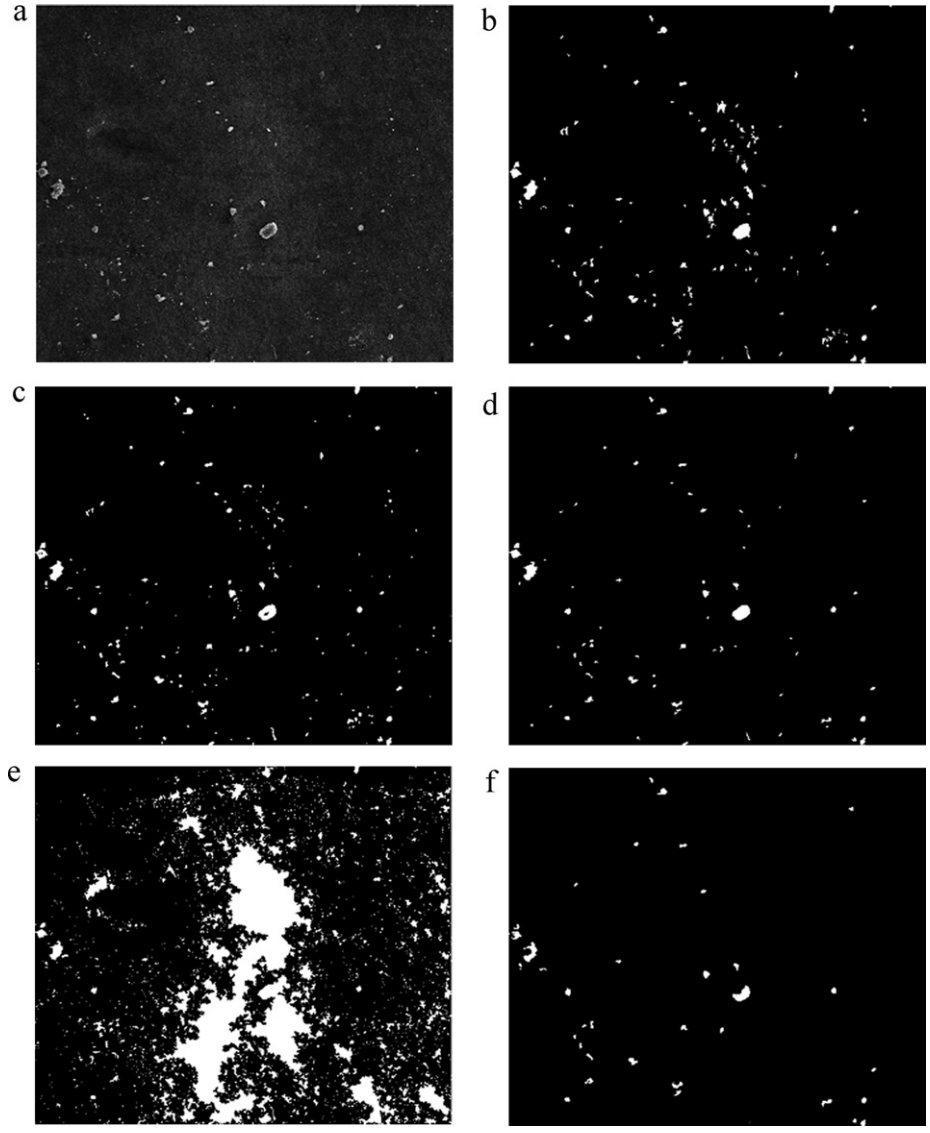

Fig. 8. Performance of segmentation algorithms used in Case 3: (a) original image, (b) Kapur threshold, (c) Minimum error threshold, (d) Otsu threshold (predicted by GLCM-based SVM), (e) Rosin threshold, (f) Sobel method (predicted by histogram-based SVM).

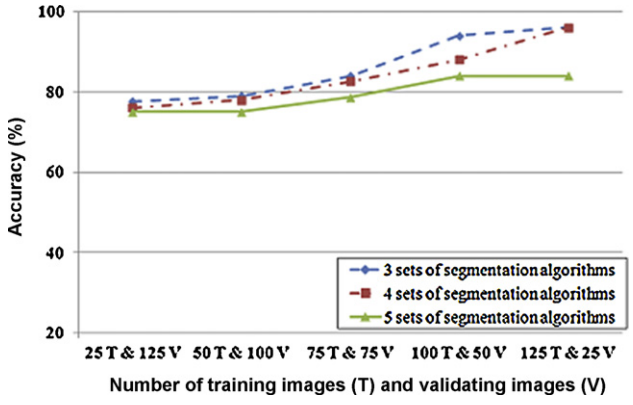

Fig. 9. Effect of varying number of training images and segmentation algorithms on the accuracy of GLCM-based SVM.

image. The Otsu threshold method predicted by the histogram-based SVM gave 47 particles. Although the Sobel algorithm showed lesser number of particles compared to the actual one, the particles segmented by this algorithm were more prominent than those by the other methods.

5.2. Statistical performance of GLCM-based and histogram-based methods

The statistical performance of the GLCM-based and the histogram-based methods are evaluated by a confusion matrix method [36]. A confusion matrix displays the number of correct and incorrect predictions made by a method when compared with the actual classifications in the test data. The matrix arrangement is in such a way that the instances of predicted classes form the columns and the numbers of actual classes are the rows of the matrix. Table 5 shows the confusion matrix of GLCM-based SVM for Case 3 with 125 testing images. The entry in row one, column one illustrates that for twenty images, Otsu was correctly predicted by GLCM-based SVM. Similarly, the entry in row two, column one indicates that

Table 4
Number of particles identified by segmentation algorithms in image of Fig. 7.

Actual number of particles	No. of particles identified by segmentation algorithms				
	Sobel	Minimum error	Otsu	Rosin	Kapur
31	20	45	47	760	95

Table 5
Confusion matrix of GLCM-based SVM classifier for Case 3.

Predicted					
Actual	Otsu	Rosin	Kapur	Sobel	Minimum threshold
Otsu	20	0	0	2	0
Rosin	2	15	2	3	1
Kapur	0	0	19	0	0
Sobel	2	0	2	23	0
Minimum threshold	3	0	6	5	20

Table 6
Performance of GLCM-based and histogram-based SVM classifiers.

No. of training images	No. of testing images	Precision		Sensitivity		Specificity		F-measure	
		Histogram	GLCM	Histogram	GLCM	Histogram	GLCM	Histogram	GLCM
25	125	0.37	0.83	0.35	0.78	0.83	0.94	0.36	0.80
50	100	0.63	0.85	0.61	0.78	0.87	0.95	0.62	0.81
75	75	0.73	0.88	0.72	0.84	0.89	0.96	0.73	0.86
100	50	0.74	0.91	0.73	0.94	0.91	0.98	0.73	0.93
125	25	0.75	0.92	0.80	0.97	0.93	0.99	0.77	0.94

Table 7
Performance comparison of SVM and ANN classifiers.

No. of training images	No. of testing images	Accuracy (%)					
		Case 1		Case 2		Case 3	
		ANN	SVM	ANN	SVM	ANN	SVM
50	10	70	90	50	80	40	80

Rosin algorithm was wrongly Otsu predicted as algorithm for two images by GLCM-based SVM.

Table 6, lists the specificity, sensitivity, precision and F-measure computed using the confusion matrix evaluated from Table 5. The sensitivity or recall is the correctly classified proportion of the test samples while the specificity gives the proportion of incorrectly classified test samples. The precision is the ratio of correctly classified images by a single segmentation algorithm among the total number of correct predictions by all the algorithms whereas the F-measure is the harmonic mean of precision and recall given in Eq. (21) [37].

$$\text{F-measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (21)$$

The sensitivity of histogram-based method was 0.35 and 0.80 for 125 and 25 testing images, respectively, while that of GLCM-based method was 0.78 and 0.97, respectively. Similarly, the specificity for histogram method varied from 0.83 to 0.93 while that of GLCM method was 0.94 to 0.99. Further, for GLCM method the largest F-measure is determined to be 0.94 while for histogram it is 0.77. Hence, we conclude that the GLCM-based SVM approach is more effective in selecting the better segmentation algorithm than the histogram-based SVM method.

5.2.1. Performance of GLCM-based SVM and ANN classifiers

The performance of two GLCM-based classifiers, namely, ANN and SVM were assessed by using the same number of training/testing images. A database containing 60 images was randomly selected and divided into 50 training and 10 testing samples. It is evident from Table 7 that the ANN gave classification efficiency of 70% for Case 1, 50% for Case 2 and 40% for Case 3 while GLCM-based SVM gave accuracy of 90%, 80% and 80%, respectively. Also, the SVM classifier required only 20 iterations while ANN required 200 iterations for the same number of training images. For Case 3, when the training images were reduced to 25, the accuracy of the SVM dropped to 80% which is still higher than that of ANN classifier (70%). This demonstrates that SVM classifiers perform better even with less number of training samples and overall are more accurate and robust than the ANN classifiers.

The performance analysis of SVM and ANN methods are determined using the confusion matrix for Case 1, Case 2 and Case 3. Table 8 shows the specificity, sensitivity, precision and F-measure computed using the confusion matrix for ten testing images. For Case 1, the sensitivity of SVM and ANN classifiers was determined as 0.93 and 0.64 respectively while the specificity was 0.96 and 0.83 respectively. Further, the largest F-measure determined by SVM is 0.90 while ANN gave 0.66 value for Case 1. Hence, our proposed method outperforms ANN method.

In future, the feature extraction capabilities of GLCM and histogram techniques can be integrated to further enhance the performance of the SVM method. The feature selection process in the GLCM method forms an important step, especially when the number of observations is relatively small compared to the number of features, and can be improved by additional pre-processing.

Table 8
Performance analysis of SVM and ANN methods derived from the confusion matrix.

No. of segmentation algorithms	Precision		Sensitivity		Specificity		F-measure	
	ANN	SVM	ANN	SVM	ANN	SVM	ANN	SVM
3 (Case 1)	0.68	0.88	0.64	0.93	0.83	0.96	0.66	0.90
4 (Case 2)	0.42	0.83	0.45	0.83	0.84	0.92	0.43	0.83
5 (Case 3)	0.36	0.83	0.45	0.80	0.83	0.95	0.40	0.81

6. Conclusion

This paper demonstrates an enhanced performance in predicting the optimal segmentation algorithm by the GLCM-based SVM in comparison to the histogram-based SVM. The GLCM-based SVM achieved an accuracy of 96% compared to 80% in the histogram-based method when three segmentation algorithms are used. When used with five segmentation algorithms, the GLCM-based SVM was 88% accurate as opposed to the histogram-based approach which provided 52% accuracy. The GLCM-based SVM outperformed the histogram-based in statistical assessment by the confusion matrix method. The maximum derived F-measure parameter from the matrix is 0.94 for the GLCM-based while it is 0.77 for the histogram-based method. The performance of GLCM-based SVM was significantly better (80–90% accuracy) than that of ANN classifier (40–70% accuracy). Thus, we demonstrate that the proposed GLCM-based SVM is more accurate than the previously developed histogram-based method and the ANN method.

Acknowledgments

The authors would like to thank the U. S. Department of Agriculture and the EECS department at The University of Toledo for their financial support. The views expressed in this paper are those of the authors.

References

[1] Particulate Matter, [Online document], Available: http://www.epa.gov/pm/(accessed 06/22/2011).
[2] A. Bhat, A. Kumar, Particulate characteristics and emission rates during the injection of class B biosolids into an agricultural field, Sci. Total Environ. 414 (2012) 328–334 (In review).
[3] J.D. Yanosky, P.L. Williams, D.L. MacIntosh, A comparison of two direct-reading aerosol monitors with the federal reference method for $PM_{2.5}$ in indoor air, Atmos. Environ. 36 (2002) 107–113.
[4] R.L. Davison, D.F.S. Natusch, J.R. Wallace, C.A. Evans, Trace elements in fly ash: dependence of concentration on particle size, Environ. Sci. Technol. 8 (1974) 1107–1113.
[5] H.Y. Tong, F.W. Karasek, Quantitation of polycyclic aromatic hydrocarbons in diesel exhaust particulate matter by high-performance liquid chromatography fractionation and high-resolution gas chromatography, Anal. Chem. 56 (1984) 2129–2134.
[6] K.K. Shandilya, A. Kumar, Physical characterization of fine particulate matter inside the public transit buses fueled by biodiesel in Toledo, Ohio, J. Hazard. Mater. 190 (2011) 508–514.
[7] H.V. Malderen, R.V. Grieken, N.V. Bufetov, K.P. Koutzenogii, Chemical characterization of individual aerosol particles in central Siberia, Environ. Sci. Technol. 30 (1995) 312–321.
[8] K. Mogireddy, V. Devabhaktuni, A. Kumar, P. Aggarwal, P. Bhattacharya, A new approach to simulate characterization of particulate matter employing support vector machines, J. Hazard. Mater. 186 (2011) 1254–1262.
[9] K.S. Fu, J.K. Mui, A survey on image segmentation, Pattern Recognit. 13 (1981) 3–16.
[10] F.H. Kong, Image retrieval using both color and texture features, Int. Conf. Mach. Learn. Cybern. 4 (2009) 2228–2232.
[11] R.M. Haralick, K. Shanmugam, H. Dinstein, Textural features for image classification, IEEE Trans. Syst. Man Cybern. 3 (1973) 610–621.
[12] J.S. Weszka, C.R. Dyer, A. Rosenfeld, A comparative study of texture measures for terrain classification, IEEE Trans. Syst. Man Cybern. 6 (1976) 269–285.
[13] R.W. Conners, C.A. Harlow, A theoretical comparison of texture algorithms, IEEE Trans. Pattern Anal. Mach. Intell. 2 (1980) 204–222.
[14] M.R. Chandraratne, Comparison of three statistical texture measures for lamb grading, IEEE Int. Conf. Ind. Inf. Syst. (2007) 513–518.
[15] M. Mustafa, M.N. Taib, Z.H. Murat, S. Lias, GLCM texture feature reduction for EEG spectrogram image using PCA, IEEE Conf. Res. Dev. (2010) 426–429.
[16] M. Seetha, I.V. Muralikrishna, B.L. Deekshatulu, B.L. Malleswari, Nagaratna, P. Hedge, Artificial neural network and other methods of image classification, J. Theor. Appl. Inf. Technol. (2008) 1039–1053.
[17] C.J.C. Burges, A tutorial on support vector machines for pattern recognition, Data Min. Knowl. Disc. 2 (1998) 121–167.
[18] O. Chapelle, P. Haffner, V.N. Vapnik, Support vector machines for histogram-based image classification, IEEE Trans. Neural Networks 10 (1999) 1055–1064.
[19] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Addision-Weseley Publication Company, Massachusetts, 1992.
[20] S.S. Varshney, N. Rajpal, R. Purwar, Comparative study of image segmentation techniques and object matching using segmentation, Proc. Int. Conf. Methods Models Comput. Sci. (2009) 1–6.
[21] J.N. Kapur, P.K. Sahoo, A.K.C. Wong, A new method for gray-level picture thresholding using the entropy of the histogram, Comput. Vision Graphics Image Process. 29 (1985) 273–285.
[22] P.L. Rosin, Unimodal thresholding, Pattern Recognit. 34 (2001) 2083–2096.
[23] N. Otsu, A threshold selection method from gray-level histogram, IEEE Trans. Syst. Man Cybern. 9 (1979) 62–66.
[24] P.K. Sahoo, S. Soltani, A.K.C. Wong, Y.C. Chen, A survey of thresholding techniques, Comput. Vision Graphics Image Process. 41 (1988) 233–260.
[25] R. Jain, R. Kasturi, B.G. Schunck, Machine Vision, McGraw-Hill Publications, New York, 1995.
[26] R.S. Choras, Image feature extraction techniques and their applications for CBIR and biometrics systems, Int. J. Biol. Biomed. Eng. 1 (2007) 6–16.
[27] N. Sharma, A.K. Ray, S. Sharma, K.K. Shukla, S. Pradhan, L.M. Aggarwal, Segmentation and classification of medical images using texture-primitive features: application of BAM-type artificial neural network, J. Med. Phys. 33 (2008) 119–126.
[28] M. Tuceryan, A.K. Jain, Texture analysis, in: C.H. Chen, L.F. Pau, P.S.P. Wang (Eds.), The Handbook of Pattern Recognition and Computer Vision, 2nd edition, World Scientific Publishing Co., 1998, pp. 207–248.
[29] C. Sun, W.G. Wee, Neighboring gray level dependence matrix for texture classification, Comput. Vision Graphics Image Process. 23 (1983) 341–352.
[30] D. Gadhkari, Image quality analysis using GLCM, MSc. Thesis, B.S.E.E. University of Pune, 2000.
[31] C. Kavitha, M.B. Rao, B.P. Rao, A. Govardhan, Image retrieval based on local histogram and texture features, Int. J. Comput. Sci. Inf. Technol. 2 (2011) 741–746.
[32] C. Cortes, V. Vapnik, Support vector networks, Mach. Learn. 20 (1995) 273–297.
[33] T. Schmah, G. Yourganov, R.S. Zemel, G.E. Hinton, S.L. Small, S.C. Strother, Comparing classification methods for longitudinal fMRI tudies, J. Neural. Comput. 22 (2010) 2729–2762.
[34] O. Chapelle, V. Vapnik, Model selection for support vector machines, Adv. Neural Inf. Process. Sys. 12 (2000).
[35] A. Akkala, V. Devabhaktuni, A. Kumar, D. Bhatt, Development of an ANN interpolation scheme for estimating missing radon concentrations in Ohio, The Open Environ. Eng. J. 4 (2011) 21–31.
[36] T.C.W. Landgrebe, R.P.W. Duin, Efficient multiclass ROC approximation by decomposition via confusion matrix perturbation analysis, IEEE Trans. Neural Network 5 (2008) 2044–2052.
[37] D.M. Dziuda, Data Mining in Genomics and Proteomics: Analysis of Gene and Protein Expression Data, John Wiley and Sons, Inc., New Jersey, 2010.