GLCM Feature-Based Texture Image Classification Using Support Vector Machine

R. Anand (ORCID)**, T. Shanthi** (ORCID)**, R. S. Sabeenian** (ORCID)**, M. E. Paramasivam** (ORCID)**, and K. Manju** (ORCID)

Abstract Texture is one of the important characteristics of an image, and its features can be used to identify specific region of interest from the image. Texture features can be extracted for any kind of images like RGB, monochrome, aerial, satellite images. This chapter describes an approach for texture image classification based on Gray-level co-occurrence matrix (GLCM) features and machine learning algorithm like support vector machine. Ten different GLCM features were extracted and fed as input to support vector machine for classification. The proposed method is trained and tested with dataset collected from center for image analysis, Swedish University. In recent days, machine learning (ML) methods are highly used to mimic the complex mathematical expressions of texture features. The main objective of this chapter is to demonstrate the state of the art of ML models in texture image prediction and to give insight into the most suitable models.

Keywords Support vector machine · Texture image · Machine learning · Confusion Matrix · Gray-level co-occurrence matrix

1 Introduction

Textures are complex visual patterns, composed of spatially organized entities that have characteristic brightness, color, shape, and size. Texture is a description of the spatial arrangement of color or intensities in an image or selected region of an image. Image texture is a function of spatial variation of pixel intensities. Texture is homogenous for human visual system. Some texture descriptions are mainly used in many real-time applications; these features are contrast, brightness, likeness, regularity, roughness etc., Texture involves spatial distribution of gray-level image,

R. Anand (✉) · T. Shanthi · R. S. Sabeenian · M. E. Paramasivam · K. Manju
Department of Electronics and Communication Engineering, Sona College of Technology, Salem, Tamil Nadu, India
e-mail: anand.r@sonatech.ac.in

143

different levels of resolutions, usually extracted form single band. Texture features from different bands (multiple bands) of the image are generally different and have capability of discriminating different land use and land cover types. Image segmentation using texture features is partitioning an image into regions that have homogenous properties with respect to texture. Image classification using texture features is assigning a label to an unknown sample image or a portion of image with a known texture class. Texture synthesis is building a model of image texture, which can be used for generating the texture.

2 Literature Survey

2.1 Statistical Approaches

Images with varying textures have certain characteristics that can be extracted statistically. Generally, statistical approaches have four prevalent methods like Gray-Level Co-occurrence Matrix (GLCM), Histogram Method, Auto-Correlation Method, and Morphological operation. These four methods have both advantages and disadvantages. Out of these four methods, this chapter adopts Gray-level co-occurrence matrix (GLCM) from which ten different features are extracted. These features are elaborated in the next section, and the literature on statistical approaches is shown in Table 1.

Table 1 Strength and weakness of statistical approach methods for texture image classifications

| Methods	Strengths	Weakness
Gray-level co-occurrence Matrix (GLCM) [8]	• Spatial relationship of pixels with ten different statistical computations. • (contrast, energy, homogeneity, mean, standard deviation, entropy, RMS, variance, smoothness, IDM) • Accuracy rate will be high.	• High computational time. • Optimum movement vector is problematic. • It requires feature selection procedure. • Accuracy depends on the offset rotation.
Histogram Method [9]	• Less computations. • Invariant to conversion and rotation. • Mathematically solvable.	• Sensitive to noise. • Low recognition rate.
Auto-correlation Method [7]	• It overcomes illumination distortion and it is robust to noise. • Low computational complexity.	• Real-time applications for large images need high computations. • Not suitable for all kinds of textures.
Morphological operation [6]	• Good efficient aperiodic image texture.	• Morphological operation cannot be applicable for periodic images.
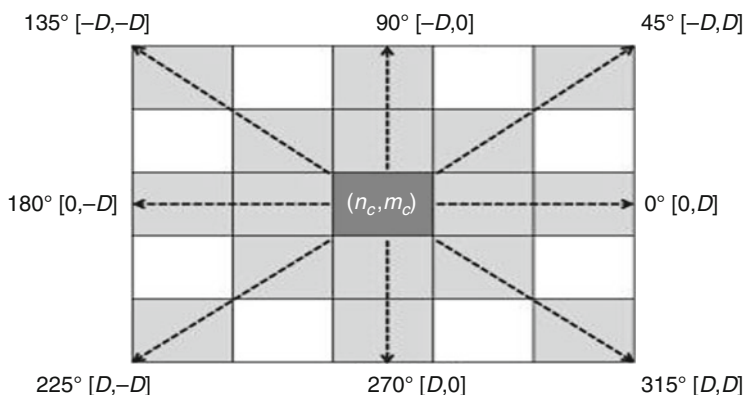
Fig. 1 Co-occurrence matrix

2.2 GLCM

The Gray-Level Co-occurrence Matrix method—likewise often named as the Spatial Gray-Level Dependence Matrix (SGLDM) Method. This method is based on the intensity values of the image. The features when fed to the machine learning algorithm perform the classification effectively [1, 2].

The second-order statistics are attained by considering a set of pixels related to each other in positive three dimensions. The Gray-Level Co-occurrence matrices provide rare mathematical statistics on the texture. Texture feature calculation uses the content of the GLCM to give a measure of the variation in intensity at the pixel of interest. GLCM matrix is a square matrix that has the same number of rows and same number of columns. GLCM matrix is N × N matrix, where N denotes the number of possible gray levels in an image. For example, a three-bit image will have a GLCM matrix of size 8 × 8. The rows and columns are the gray values (0–7). GLCM matrix of image depends on the direction and offset values. The direction can be anyone among the eight possible directions as shown in Fig. 1. The offset represents distance between pixels.

This chapter utilizes ten different features that are computed from GLCM Matrix [13]. These ten features are elaborated as follows.

2.2.1 Entropy

Entropy measures the disorder or complexity of an image. If the entropy is large when the image is not texturally uniform and if the texture is complex, then entropy becomes high. Energy and entropy are inversely proportional to each other; it means if entropy is high, energy will be low and vice versa. Entropy can be calculated for image $f(x, y)$ using the following equation:

$$\text{Entropy (ENT)} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) * \log(f(x, y))$$

2.2.2 Contrast

Contrast measures the spatial frequency of an image and different moment of GLCM. It is the difference between the highest and the lowest values of a contiguous set of pixels. Contrast can be calculated for image $f(x, y)$ using the following equation:

$$\text{Contrast} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} (x - y)^2 f_{xy}$$

2.2.3 Mean

Mean is the normal value of the whole amount of data. Mathematically,

$$\text{Mean } (\mu i) = \frac{\sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y)}{M x N}$$

2.2.4 Standard Deviation

The standard deviation is the square root of variance divided by the entire. number of samples. Mathematically, it is expressed as

$$SD = \sqrt{\frac{\sum |x - \mu|^2}{N}}$$

2.2.5 Variance

This statistic measures the heterogeneity, and it is strongly correlated with first-order statistical variable such as standard deviation. Variance can be calculated for image $f(x, y)$ using the following equation:

$$\text{Variance} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} (i - \mu)^2 f_{xy}$$

where 'μ' is the mean of an input image.

2.2.6 Homogeneity

Homogeneity is the consistency in the arrangement of an input image $f(x, y)$. If the arrangement of an input image follows a regular pattern, then it is said to be homogenous. Mathematically, it can be expressed by the following equation:

$$\text{Homogenous } f(x, y) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \frac{f_{xy}}{1 + |x - y|^2} \Big)$$

2.2.7 RMS Contrast

Root Mean Square (RMS) measures the standard deviation of the pixel intensities. It does not depend upon any angular frequency or spatial distribution of contrast of an input image. Mathematically, it can be expressed as

$$\text{RMS Contrast} = \sqrt{\frac{1}{MN} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \left(I_{ij} - \overline{I}\right)^2}$$

where $\acute{I}$ is the normalized pixel intensity values between 0 and 1.

2.2.8 Smoothness

If the input image $f(x, y)$ is highly correlated between the adjacent pixels, then we can say the image is autocorrelated (the autocorrelation of input data with itself after shifting one pixel). The autocorrection of input image can be calculated by the following equation:

$$\text{autocorrelation } f(x, y) = \iint f(x, y) * \left(f(x + \overline{x})(y + \overline{y}) \, dx \, dy\right)$$

2.2.9 Inverse Difference Moment (IDM)

IDM measures the local similarity between of an image. It has the feature of measuring the closeness and distribution of the GLCM elements with respect to GLCM diagonal. It also has the property to determine the texture of the image. The frequency of co-occurrence of set of pixels is improved when their values of gray level are close enough, and this increases the IDM value.

$$\text{Inverse Difference Moment (IDM)} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \frac{f_{xy}}{1 + |x - y|^2}$$

3   Proposed Method

Texture features of an image are calculated considering only one band at a time.
Channel information can be consolidated using PCA before calculating texture
features. Texture features of an image can be used for both supervised and unsu-
pervised image classification. In this chapter, we have used support vector machine
(SVM) for texture image classification. This method falls under the category of
supervised machine learning [10]. Support vector machine was introduced in 1992
as a supervised machine leaning algorithm. This algorithm gained its popularity
because of its higher accuracy rate and minimum error rate. SVM is one of the
best example for "Kernel Method' which is the key area of machine learning that
is shown in Fig. 2. The idea behind SVM is to make use of nonlinear mapping
function φ, that transforms data in input space to data in feature space in such way
that it becomes a linearly separable [3]. The SVMs then automatically discover the
optimal separating hyperplane, which is nothing but a complex decision surface.
The equation of hyperplane is derived from a line equation $y = ax + b$. However,
even though hyperplane is a line, its equation is shown below [11, 12], where "w
and x" are the vectors, and it can be computed by dot matrix of these two vectors
[5].

$$w\hat{}T\ x = 0$$

Any hyperplane can be written as the set of points, x satisfying the equation
$w.\ x + b = 0$. A optimum hyperplane separates the data points with minimum loss
function. Two such hyperplanes are chosen and based on the values obtained; they
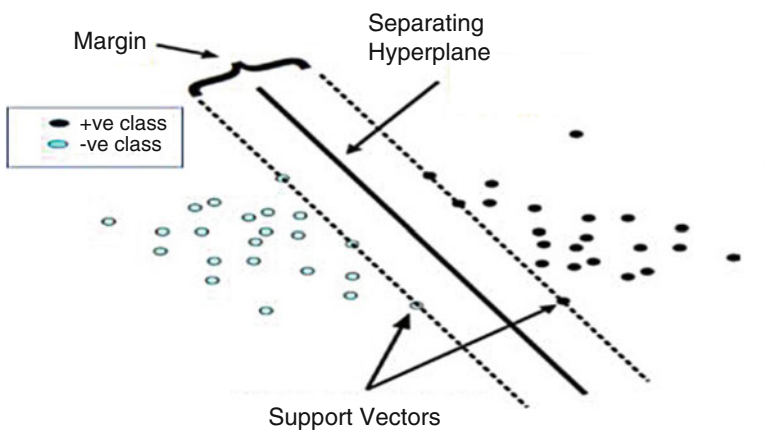are classified as class 1 and class 2 as given below.

Fig. 2 Separating hyperplane in SVM

$$w.x + b \geq 1 \text{ for } x\_i \text{ having class } 1$$

$$w.x + b \leq -1 \text{ for } x\_i \text{ having class } 2$$

Here, optimizing problem may occur because the goal is to maximize the margin among all the possible hyperplanes meeting the constraints. The hyperplane with the small ||w|| is chosen because of the biggest margin it provides. The pseudo codes for optimizing problem are given below:

$$\text{Minimize } in (w, b), \|w\|, \text{ subject } to \ y\_i \ (w.x + b) \geq 1$$

The solution to the above equation is computing the values of (w, b), with minimum possible margin. The equation that satisfies the constraints will be considered as the equation of the optimal hyperplane [14, 15].

3.1 Dataset Description

The dataset from center for image analysis in Swedish university of agriculture sciences [4], Uppsala University has been used in this chapter. Totally, 4480 images of 28 different texture classes were taken using Canon EOS 550d DSLR camera, which is shown in Fig. 3. Each texture class has around 160 images, in that 112 images are used for training and 48 images are used for testing which is shown in table [2]. Fig. 4 shows the complete flow chart of the proposed method for texture image classifications using GLCM features. First step, the segmented image is resized to [576 x 576] (Table 2).

The performance of the proposed system is measured in terms of sensitivity, specificity, accuracy, precision, false positive rate and false negative rate, which is shown in Fig. 5. The sensitivity and specificity are important measures in classification. The accuracy of the system represents the exactness of the system with respect to classification. The precision indicates the ability of the system to produce the same result when tested at different measurements. The overall classification accuracy of the proposed system is around 99.4% with the precision of 92.4% . The false negative rate and false positive rate is very minimum in the range of 0.003 and 0.085, respectively. The sensitivity and specificity of the system is around 91.5% and 99.7%.The texture classes 2,4,5,7,9,12, and 19 have been classified with good accuracy and precision when compared to other classes.

The GUI of the proposed system is shown in Fig. 6 (Table 3).
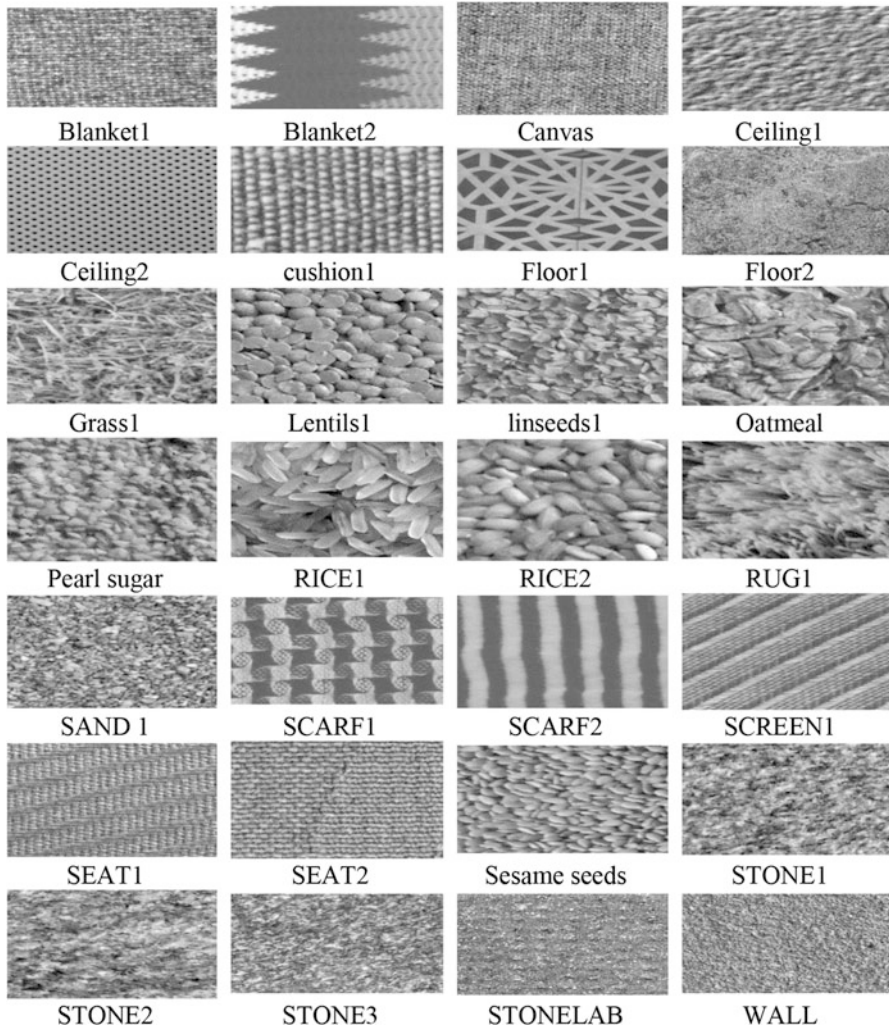
Fig. 3 Sample images from 28 different texture images
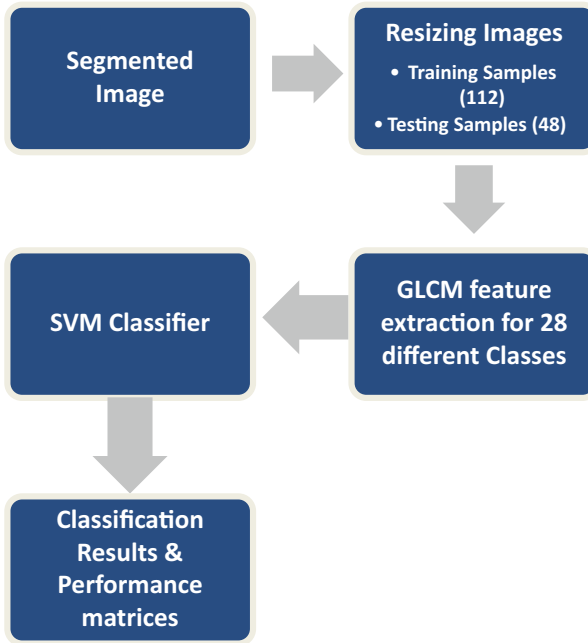
Fig. 4 Flow chart for produced texture image classifications

Table 2 Dataset descriptions

Class label	Training samples	Testing samples	Total samples
Blanket1	112	48	160
Blanket2	112	48	160
Canvas	112	48	160
Ceiling1	112	48	160
Ceiling2	112	48	160
cushion1	112	48	160
Floor1	112	48	160
Floor2	112	48	160
Grass1	112	48	160
Lentils1	112	48	160
linseeds1	112	48	160
Oatmeal	112	48	160
Pearl sugar	112	48	160
RICE1	112	48	160
RICE2	112	48	160

Table 2 (continued)

Class label	Training samples	Testing samples	Total samples
RUG1	112	48	160
SAND 1	112	48	160
SCARF1	112	48	160
SCARF2	112	48	160
SCREEN1	112	48	160
SEAT1	112	48	160
SEAT2	112	48	160
Sesame seeds	112	48	160
STONE1	112	48	160
STONE2	112	48	160
STONE3	112	48	160
STONELAB	112	48	160
WALL	112	48	160

Fig. 5 Prediction using support vector machine

Fig. 6 Performance metrics evaluation for texture image classification

Table 3 Confusion matrix for 28 classes

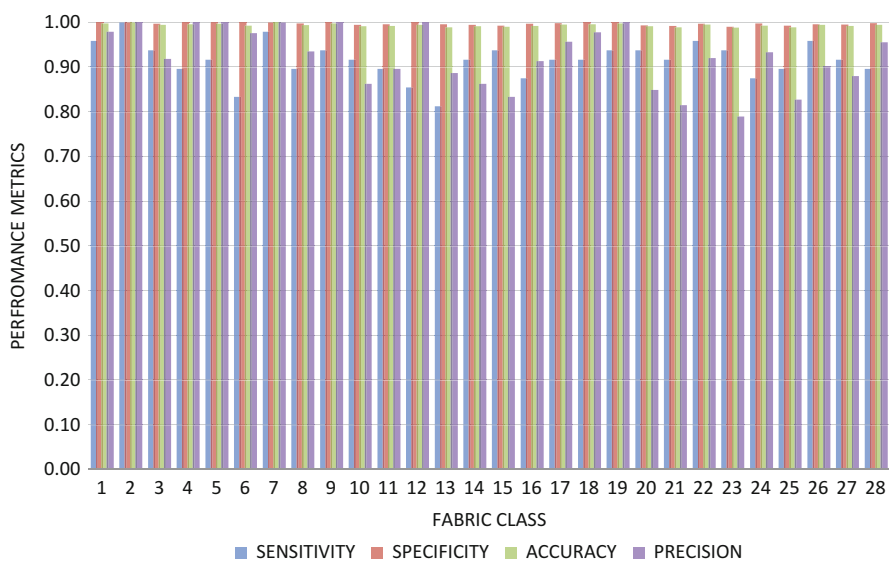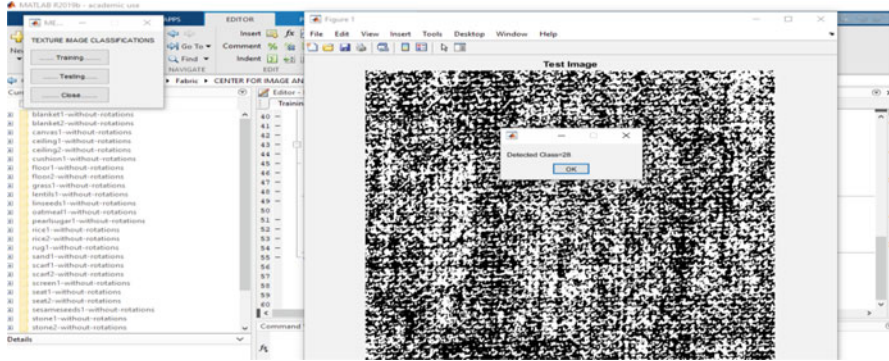| Predicted class / Actual Class	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
1	46	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
2	0	48	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	45	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
4	1	0	0	43	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	2	0
5	0	0	0	0	44	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	1	0
6	0	0	0	0	0	40	0	0	0	0	0	0	0	3	1	0	0	0	0	0	0	0	0	4	0	0	0	0
7	0	0	0	0	0	0	47	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	1	0	43	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
9	0	0	0	0	0	0	0	0	45	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	1	0	0	0
10	0	0	0	0	0	0	0	0	0	44	0	0	0	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	43	0	0	2	1	0	0	0	0	0	0	0	2	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	4	0	41	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	3	0	39	0	0	0	2	0	0	0	0	0	3	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	2	0	0	44	2	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	1	0	0	0	45	2	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	42	0	0	0	0	0	0	0	0	2	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	44	0	0	0	3	1	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	44	0	0	2	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	45	2	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45	3	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	44	1	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	46	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	45	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	42	4	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	43	3	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	46	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	44	2
28	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	43

4 Conclusion

Texture of an image is a description of the spatial arrangement of color or intensities in the image. Texture of image can be used to categorize the image into several classes. Ample number of texture features can be computed mathematically and used for image analysis. The method proposed in this chapter combines the texture features computed from GLCM matrix with that of the standard machine learning algorithm for image classification. The overall classification accuracy of the proposed system is around 99.4% with the precision of 92.4%. The classification accuracy of the system can be further improved by increasing the size of the dataset.

References

1. Anand, R., Shanthi, T., Nithish, M. S., & Lakshman, S. (2020). Face recognition and classification using GoogleNET architecture. In K. Das, J. Bansal, K. Deep, A. Nagar, P. Pathipooranam, & R. Naidu (Eds.), *Soft computing for problem solving. Advances in intelligent systems and computing* (Vol. 1048). Singapore: Springer.
2. Veni, S., Anand, R., & Vivek, D. (2020). Driver assistance through geo-fencing, sign board detection and reporting using android smartphone. In K. Das, J. Bansal, K. Deep, A. Nagar, P. Pathipooranam, & R. Naidu (Eds.), *Soft computing for problem solving. Advances in intelligent systems and computing* (Vol. 1057). Singapore: Springer.
3. Veni, S., Anand, R., & Vivek, D. (2020). Driver assistance through geo-fencing, sign board detection and reporting using android smartphone. In K. Das, J. Bansal, K. Deep, A. Nagar, P. Pathipooranam, & R. Naidu (Eds.), *Soft computing for problem solving. Advances in intelligent systems and computing* (Vol. 1057). Singapore: Springer.
4. G. Kylberg. The Kylberg texture dataset v. 1.0, Centre for Image Analysis, Swedish University of Agricultural Sciences and Uppsala University, External report (Blue series) No. 35. Available online at: http://www.cb.uu.se/gustaf/texture/
5. Anand, R., Veni, S., & Aravinth, J. (2016). An application of image processing techniques for detection of diseases on brinjal leaves using k-means clustering method. *2016 International Conference on Recent Trends in Information Technology* (ICRTIT). IEEE.
6. Sabeenian, R. S., & Palanisamy, V. (2009). Texture-based medical image classification of computed tomography images using MRCSF. *International Journal of Medical Engineering and Informatics, 1*(4), 459.
7. Sabeenian, R. S., & Palanisamy, V. (2008). Comparison of efficiency for texture image classification using MRMRF and GLCM techniques. *Published in International Journal of Computers Information Technology and Engineering (IJCITAE), 2*(2), 87–93.
8. Haralick, Robert, M., Shanmugam, K., & Dinstein, H. (1973). Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics, 6*, 610–621.
9. Varma, M., & Zisserman, A. (2005). A statistical approach to texture classification from single images. *International Journal of Computer Vision, 62*(1–2), 61–81.
10. Suykens, Johan, A. K., & Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters, 9.3*, 293–300.
11. Scholkopf, B., & Smola, A. J. (2001). *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press.
12. Bennett, K. P., & Demiriz, A. (1999). Semi-supervised support vector machines. In *Advances in neural information processing systems* (pp. 368-374).
13. Shanthi, T., Sabeenian, R. S., & Anand, R. (2020). Automatic diagnosis of skin diseases using convolution neural network. *Microprocessors and Microsystems, 103074*.
14. Shanthi, T., & Sabeenian, R. S. (2019). Modified Alexnet architecture for classification of diabetic retinopathy images. *Computers and Electrical Engineering, 76*, 56–64.
15. Hall-Beyer, M. (2000). GLCM texture: a tutorial. *National Council on Geographic Information and Analysis Remote Sensing Core Curriculum, 3*.