Neurocomputing 000 (2017) 1–12

Contents lists available at ScienceDirect

Neurocomputing

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

Gaussian derivative models and ensemble extreme learning machine for texture image classification

Yan Song[a], Shujing Zhang[b], Bo He[a,*], Qixin Sha[a], Yue Shen[a], Tianhong Yan[c], Rui Nian[a], Amaury Lendasse[d]

[a] School of Information Science and Engineering, Ocean University of China, Qingdao, Shandong 266000, China
[b] College of Vocation Technology, Hebei Normal University, Shijiazhuang, Hebei 050024, China
[c] School of Mechanical and Electrical Engineering, China Jiliang University, Hangzhou 310018, China
[d] Department of Mechanical and Industrial Engineering and the Iowa Informatics Initiative, The University of Iowa, Iowa City, IA 52242-1527, USA

ARTICLE INFO

Article history:
Received 29 September 2016
Revised 6 January 2017
Accepted 8 January 2017
Available online xxx

Keywords:
Gaussian derivative models
Extreme learning machine
Ensemble extreme learning machine
Texture classification
Gabor filters

ABSTRACT

In this paper, we propose an innovative classification method which combines texture features of images filtered by Gaussian derivative models with extreme learning machine (ELM). In the texture image classification, feature extraction is a very crucial step. Thusly, we use linear filters consisting of two Gaussian derivative models, difference of Gaussian (DOG) and difference of offset Gaussian (DOOG), to detect texture information of images. Besides, ensemble extreme learning machine (E^2LM) is proposed to reduce the randomness of original ELM and used as the classifier in this paper. We evaluate the performance of both the texture features and the classifier E^2LM by using three datasets: Brodatz album, VisTex database and Berkeley image segmentation database. Experimental results indicate that Gaussian derivative models are superior to Gabor filters, and E^2LM outperforms the support vector machine (SVM) and ELM in classification accuracy.

1. Introduction

Image classification is an important topic in image processing, which can assign each pixel of an image to one class according to the characteristics of their textures. Meanwhile, the texture image classification plays a crucial role in the industrial product identification, the remote sensing image processing and some other fields of science. And in the texture image classification, the texture features extraction (or selection) and the choice of a classifier are two main factors to be considered.

Texture features extraction determines whether or not the texture classification method has a good result. Many approaches have been proposed so far to extract texture information from images, among which Gauss Markov Random Field (GMRF) [1], Local Binary Patterns (LBP) [2,3], Gray Level Co-occurrence Matrix (GLCM) [4] and Gabor filters [5–7] are typical and widely used. Sparse coding has been used in image information extraction in [8–11]. These algorithms extract information from images, project images into feature spaces and provide different approaches for texture image classification.

In this paper, we propose an innovative method of pixel-level texture image classification based on texture features extracted from images which are filtered by two linear filters, Difference of Gaussian (DOG) and Difference of Offset Gaussian (DOOG) [12–18]. We use DOG filters to describe spotted regions and DOOG filters to describe changes of different orientations. Each texture image has a set of filtered images, which are used as feature vectors of every pixel in the texture classification experiments.

The choice of a classifier is another crucial factor of texture image classification. In this paper, extreme learning machine (ELM) [19], which has one hidden layer and one output layer, is proposed for image classification using the features mentioned above. The advantages of ELM are that the input nodes and hidden layer biases of ELM can be assigned randomly and it learns faster while achieving high accuracy rate. In [20], ELM was used to perform classification of hyperspectral imagery (HSI) by exploiting texture features of LBPs. The classification results are superior to some traditional alternatives. In [21], kernel ELM (KELM) was used as the classifier to carry out spectral–spatial classification and the classification results were better than that of support vector machine (SVM). In the meantime, ELM is introduced to classify complex dynamic texture in Ref. [22] and show better effect. Besides, ELM has been extensively studied in object recognition [23].

* Corresponding author.
E-mail address: bhe@ouc.edu.cn (B. He).

http://dx.doi.org/10.1016/j.neucom.2017.01.113
0925-2312/

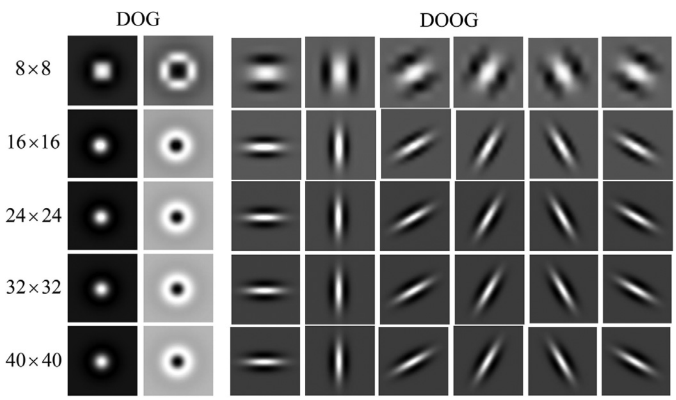

Fig. 1. The DOG filters and DOOG filters.

To improve the performance of ELM, ensemble extreme learning machine (E^2LM) is proposed to eliminate the randomness of traditional ELM. For E^2LM, k-fold cross validation [24] is used to choose the optimal parameters of every classifier and majority voting [25] is applied in the determination of the classification results.

Therefore, there are two primary contributions in this paper. First, we extract texture features from images filtered by DOG and DOOG filters with different scales and different orientations. These features are compared with Gabor filters in the experiments. Second, we use E^2LM as the classifier in image classification and compare the classification results of E^2LM with those of ELM and support vector machine (SVM) [26,27] using three different datasets.

The rest of this paper begins with an introduction of ELM in Section II. Section III gives a detailed description of Gaussian derivative models. Our proposed classifier E^2LM is presented in Section IV. Section V reports and discusses the experimental results. Section VI concludes.

2. Brief review of extreme learning machine

ELM was proposed by Huang et al. [19] and it can be seen as generalized single hidden layer feedforward neural networks (SLFNs). ELM can assign the parameters of the hidden layer randomly without any iterative tuning. Besides, all the parameters of the hidden layer in ELM are mutual independence. There are two important parts for ELM: first, to calculate the output weights; second, to determine the node number optimizing networks.

There are N training samples $(x_i, t_i)_{i=1,\ldots,N}$. $x_i \in \mathrm{R}^n$ consists of the input data of n dimensions, and $t_i \in \{1, \ldots, m\}$ is a label data of m dimensions. The hidden nodes of ELM are set to $\tilde{N}$ and the output function is $G(a_i, b_i, x)$. So ELM model can be described as

$$f_{\tilde{N}}(x_j) = \sum_{i=1}^{\tilde{N}} \beta_i G(a_i, b_i, x_j) = t_j,\ j = 1, \ldots, N, \tag{1}$$

where the input weight $A = [a_1, a_2, \ldots, a_{\tilde{N}}]$ and bias $B = [b_1, b_2, \ldots, b_{\tilde{N}}]$ are parameters of hidden layer, and β_i is the output weight of ith hidden node used to connect with the output node. Equally, (1) can be written as:

$$H\beta = T, \tag{2}$$

where

$$H(a_1, \ldots, a_N, b_1, \ldots, b_{\tilde{N}}, x_1, \ldots, x_N) = \begin{bmatrix} G(a_1, b_1, x_1) & \cdots & G(a_{\tilde{N}}, b_{\tilde{N}}, x_1) \\ \vdots & \ddots & \vdots \\ G(a_1, b_1, x_N) & \cdots & G(a_{\tilde{N}}, b_{\tilde{N}}, x_N) \end{bmatrix}_{N\times\tilde{N}} \tag{3}$$

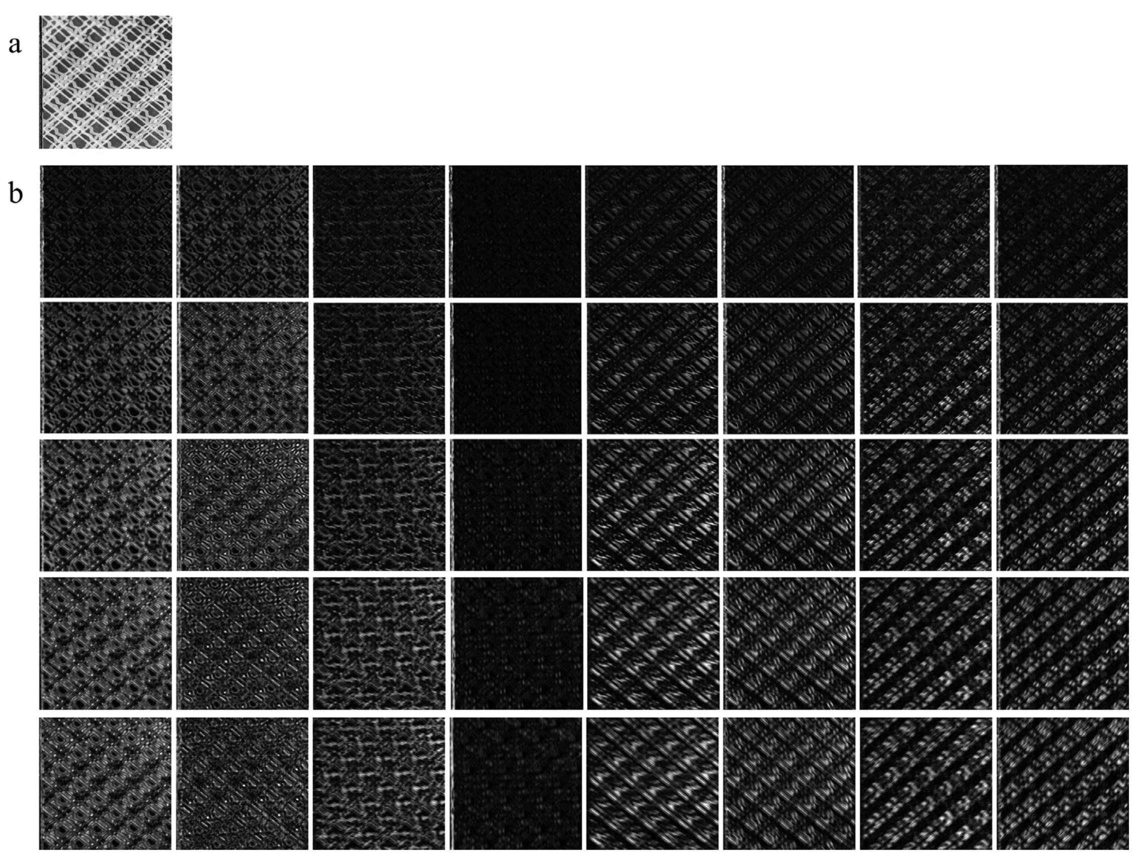

Fig. 2. (a) Original images. (b) Convolution results using DOG filters and DOOG filters.