Available online at www.ijpe-online.com

International Journal of Performability Engineering

vol. 14, no. 3, March 2018, pp. 445-452
DOI: 10.23940/ijpe.18.03.p5.445452

Colorized Image Forgery Detection based on Similarity Measurement of Gaussian Mixture Distribution

Ze Yang[a], Jianhou Gan[b,*], Juxiang Zhou[b], Bin Wen[a], and Jun Wang[b]

[a]*College of Computer Science and Technology, Yunnan Normal University, Kunming, 650500, China*
[b]*Key Laboratory of Education Informalization for Nationalities of Ministry of Education, Yunnan Normal University, Kunming, 650500, China*

Abstract

In the era of rapid development of multi-media information, forgery detection has become an important research field of digital image security. This paper proposes a new method to detect the forged image generated by deep learning. First, the feature matrix is constructed through extracting each pixel value of channels a and b in Lab color space for the real and the forged image training set, respectively, which is used to fit the Gaussian Mixture Model (GMM) distribution. Then, the Expectation Maximization (EM) adaptation algorithm is used to re-fit the GMM for test image using the obtained GMM parameter as prior information. Finally, the similarity between two GMM is calculated for forgery detection. Experiments show that the proposed method is more accurate than the traditional SVM for forgery detection.

Keywords: image tampering; forgery detection; Gauss mixed distribution; SVM

(Submitted on December 8, 2017; Revised on January 16, 2018; Accepted on February 17, 2018)

1. Introduction

With the increasing popularity of image processing software, it brings convenience to people's lives. However, it is difficult for naked eyes to distinguish the authenticity of forged digital images. As a result, it is more likely to bring a negative influence on politics, cultures, media, military affairs, laws and so on. Therefore, it is imperative to test the authenticity of images. The forgery detection of digital images is mainly divided into two kinds: the active detection and the passive detection (blind detection) [7]. The traditional active detection includes digital signature and digital watermarking technology, both of which require putting some accessional information in images in advance. In the detection, in order to judge whether the image has been tampered or not, it needs to check whether the digital watermarking or the signature digest are changed. However, some priori information may not be embedded into the image beforehand, and on the other hand, embedding may lower the image quality. Besides, these two methods will be limited by the application conditions and are incapable of putting an end to some ways of tampering. Consequently, the active detection is not a mainstream detection method in the field of forgery detection.

Nevertheless, unlike the active detection technique, passive detection needs no special requirements for being-measured images. It is unnecessary to embed any additional priori information in advance. Only the image features and the statistical information are used to identify the authenticity of the image [6], so a wider range of applications is shown. Hence, many researchers have conducted a host of deep studies on passive detection technology. So, the passive blind detection has also become a significant direction in the field of digital image security.

For target images of passive detection, there are many ways of forgery technologies. In this paper, we classify them into six categories according to the method proposed by Hany Farid et al [2]. (1) Synthesis: selecting some areas of the image and copying them to other areas of this image or other images for blocking the information in the target image or adding

* Corresponding author.
E-mail address: ganjh@ynnu.edu.cn

extra information. (2) Variants: an image is fused with other features to produce a new image. The feature points of the target image and the given feature points are usually superimposed according to a certain weight, so that the newly generated image has both two kinds of features. (3) Retouch: the details of the image are modified for effect-enhancing. (4) Strengthening: some parts of the content are emphasized by changing the local color, brightness and contrast of the image. (5) Computer generating: an image model is built through 3D modeling. The color, texture and lightness are added into the model. (6) Painting: art professionals draw images with the application of the image editing software based on their personal mastery of painting skills.

There are many new research results about these forgery technologies at home and abroad. Ye Xi presents a way of blind image tamper detection based on Radon and Fourier-Mellin transform. The proposed algorithm result is better than that of detection method based on orthogonal moments, and the white noise with mean zero was significantly higher than the robustness of the detection method based on orthogonal moments [10]. JM Yang et al. proposed a splicing image tamper detection method based on human face color temperature. Using the technology of target detection, the method in question pinpointed the faces in the image [11]. V Thirunavukkarasu et al. proposed a robust technique to discover copy-move tampering with Fast Retina Key Point Descriptor (FREAK). Suspected image is pre-processed and FREAK feature descriptors around each Harris corner points were discovered. Detected FREAK features are mapped by means of KD-Tree algorithm [8]. Junliu Zhong et al. propose an efficient Discrete Radon Polar Complex Exponential Transform (DRPCET)-based method for the extraction of the rotational and the scaling invariant features for the copy move forgery detection. The results show that the proposed method can detect the copy move region in the forgery image precisely even though the forgery regions suffered from mixed geometric distortions [12].

In this paper, a new forgery technique is used to construct forgery data sets. Richard Zhang and others' deep learning method [13] is applied to recolor images automatically. These are mainly done by designing a specific framework of a convolutional neural network, inputting a grayscale image on the input end and outputting a colorful image with a very vivid effect on the output end, which is difficult for naked eyes to discern authenticity. For the image generated by this forgery technology, none of the scholars have done a related forgery detection before. In this paper, a new algorithm is proposed to fit GMM by extracting the features of natural and forged image data sets respectively. The obtained GMM parameters are used as the priori information and the EM adaptation algorithm is applied for re-fitting the GMM. Through multiple experiments based on the constructed data sets, we calculate the similarity between two Gauss mixed distributions to detect the color forgery image generated by the deep learning. The experimental results manifest the validity of our algorithm.

2. Methodology

2.1. Construction of Gaussian Mixture Model

To illustrate the color feature of images in the real datasets and the forged dataset, a model for data points extracted from color feature of images is constructed based on the GMM. The forged dataset actually comprise color images generated on the basis of quantitative probability distribution done by Richard Zhang et al. [13] Through deep learning in Lab color space, the color information of each pixel from each image in real datasets and forged data set can be recorded as $(a, b)^T$. Therefore, each pixel point is represented by a two-dimensional vector. Assuming that there are *N* pixels in real datasets and forged dataset amounts respectively, the feature vectors of all pixels in the dataset constitute a low-level feature vector set Y in the form of a matrix of $N\times2$. The feature matrixes extracted from real datasets and forged dataset are used to fit GMM, and each Gaussian component corresponds to a continuous region of similar colors in an image from the dataset.

2.1.1. Fitting Gaussian Mixture Distribution

The value of (a, b) for any pixel of the images in the real datasets and forged dataset can be viewed as data generated through the corresponding GMM. Assuming that the number of Gaussian component included in the corresponding GMM of the image is recorded as *k*, then the probability density function of the GMM can be derived by linear addition of those Gaussian components, as shown in Equation (1):

$$P(x \mid \emptyset) = \sum_{i=1}^{k} a_i \, N_i(x|\theta_i) \tag{1}$$

Where the number of x is N, then$\{x = x_1, x_2, x_3, \dots, x_N\}$, $\emptyset$ is the parameter set of the Gaussian Mixture Distribution, a_i is the weight for the k^{th} Gaussian Mixture Distribution$(0 \le a_i \le 1)$, and $\sum_{i=1}^{k} a_i = 1$, N_i is the i^{th} component of the Gaussian Mixture Distribution. Its Gaussian probability density function is as shown in Equation (2):

$$N_i(x|\theta_i) = \frac{1}{(2\pi)^{d/2}|\sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu_i)^T\sigma^{-1}(x-\mu_i)} \quad (2)$$

where θ_i represents the parameter set of the ith component in Gaussian Mixture Distribution (μ_i, σ_i), $\mu_i$is the mean of N_i, σ_i is the covariance matrix of N_i, d is the dimension of x, The parameter set of probability density function of the corresponding GMM for images in real datasets and forged dataset is as shown in Equation (3):

$$\{(\mu_1, \sigma_1, a_1), (\mu_2, \sigma_2, a_2), (\mu_3, \sigma_3, a_3), \dots, (\mu_k, \sigma_k, a_k)\} \quad (3)$$

2.1.2. The Solution of Gaussian Mixture Distribution Parameters

The aim of fitting GMM with low-level features of images is to get the GMM parameter set. Since the low-level feature matrix x can be viewed as data generated by the corresponding GMM, with the given GMM probability density function, the maximum likelihood parameters of GMM can be calculated with the EM algorithm [4]. The EM algorithm flow is mainly divided into the following steps:

(1) In order to speed up the convergence and reduce iteration time, k-means is taken to initialize the mean μ_i, covariance σ_i and weight $a_i(1,2,3,\dots,k)$, and to calculate the logarithmic likelihood value to set conditions for convergence as shown in Equation (4):

$$ln\, p(x|\phi) = \sum_{i=1}^{N} ln\left\{\sum_{i=1}^{k} \pi_k\, N(x_i|\mu_k, \sigma_k)\right\} \quad (4)$$

(2) E-step: estimating the probability for the data to be generated by each Gaussian component. For each data x_i, the probability of being generated by *k* Gaussian composition is as shown in Equation (5):

$$r(i,k) = \frac{\pi_k N(x_i|\mu_k, \sigma_k)}{\sum_{j=1}^{k} \pi_j\, N(x_i|\mu_j, \sigma_j)} \quad (5)$$

Where $N(x_i|\mu_k, \sigma_k)$ is the posterior probability.

(3) M- step: calculating the new mean μ_i, covariance σ_i, and weight a_i(i = 1,2,3, … , k) by taking derivative on the basis of the posterior probability value with the maximum likelihood estimation as shown in Equation (6) ~ (8),

$$\mu_k = \frac{1}{\sum_{i=1}^{N} r\,(i,k)} \sum_{i=1}^{N} r\,(i,k) x_i \quad (6)$$

$$\sigma_k = \frac{1}{\sum_{i=1}^{N} r\,(i,k)} \sum_{i=1}^{N} r\,(i,k)(x_i - \mu_k)(x_i - \mu_k)^T \quad (7)$$

$$a_k = \frac{1}{N} \sum_{i=1}^{N} r\,(i,k) \quad (8)$$

(4) Repeat the iterative E-step and M-step until the likelihood function value converges.

GMM parameters obtained by fitting the color feature of images from the real datasets and forged dataset will be used in the subsequent detection.

2.2. EM - adaptation Algorithm

To obtain relevant parameters by fitting GMM with the feature matrixes extracted from images in real datasets and forged dataset, fitting of GMM for the input testing images will be done with the two sets of parameters respectively. Due to the

fact that it requires a large amount of data for a fast convergence of GMM fitting, the EM adaptation algorithm has proven to be an excellent choice to solve the limitation of the color feature of one image. Therefore, with the color feature matrix of input testing images, refitting GMM can be achieved through EM adaptation algorithm proposed by Enming Luo et al [5]. Then, the GMM parameters obtained together with a small amount of data points can fit a new GMM. The specific flow of EM adaptation algorithm is as follows: the E-step mainly compute the likelihood of $\tilde{x}_i$ conditioned on the generic parameter (a_k, μ_k, σ_k) as shown in Equation (9), from Equation (10) to (12), $(\tilde{a_k}, \tilde{\mu_k}, \tilde{\sigma_k})$ and are updated through a linear combination of the contributions from the new data and the generic parameters.

Algorithm 1: EM adaptation algorithm

Input: $\theta = \{(a_k, \mu_k, \sigma_k)\}_{k=1}^{K}$, $\{\tilde{x_1}, \ldots, \tilde{x_n}\}$
Output: Adapted parameters $\theta = \{(\tilde{a_k}, \mu_k, \tilde{\sigma_k})\}_{k=1}^{K}$
E-step: calculate k = 1,2,3,…K; i =1,2,3,…n

$$r(i,k) = \frac{a_k N\left(\tilde{x_i} \middle| \mu_k, \sigma_k\right)}{\sum_{j=1}^{k} a_j N\left(\tilde{x_i} \middle| \mu_j, \sigma_j\right)}, \quad n_k = \sum_{i=1}^{n} r(i,k) \tag{9}$$

M-step: calculate k = 1,2,3,…,K

$$\tilde{\mu_k} = \alpha_k \frac{1}{n_k} \sum_{i=1}^{n} r\,(i,k)\tilde{x_i} + (1-\alpha_k)\mu_k \tag{10}$$

$$\tilde{\sigma_k} = \alpha_k \frac{1}{n_k} \sum_{i=1}^{n} r\,(i,k)(\tilde{x_i} - \tilde{\mu_k})(\tilde{x_i} - \tilde{\mu_k})^T + (1-\alpha_k)(\sigma_k + (\mu_k - \tilde{\mu_k})(\mu_k - \tilde{\mu_k})^T) \tag{11}$$

$$\tilde{a_k} = \alpha_k \frac{1}{n}(1-\alpha_k)a_k \tag{12}$$

Where $\theta = \{(a_k, \mu_k, \sigma_k)\}_{k=1}^{K}$ stands for the parameter set obtained by fitting GMM with data points extracted from color feature of the images in real datasets or forged dataset. $\{\tilde{x_1}, \ldots, \tilde{x_n}\}$ is the data point extracted from color features of the testing image, α_k is the proportional coefficient $\alpha_k = n_k/(n_k + \rho)$. Experiment shows that when ρ is between 8 and 20 , the optimal results can be achieved. Here, $\rho = 15$ is taken, $\theta = \{(\tilde{a_k}, \tilde{\mu_k}, \tilde{\sigma_k})\}_{k=1}^{K}$ is obtained by fitting a new GMM with input testing images.

2.3. GMM similarity measurement with Monte Carlo simulation

In order to compare the similarity between the GMM fit by real datasets and forged datasets and the GMM fit by each testing image, Monte Carlo Simulation [3] is adopted to calculate Dmc1 and Dmc2 due to the absence of closed-form expression for the similarity between GMMs. According to Monte Carlo Simulation, x_i is taken from a probability density function F, such as $E_f[logF(x_i)/G(x_i)] = D(F||G)$. The sample data points $\{x_i\}_{i=1}^{n}$ follow independent and identical distribution. The similarity between the GMMs can be calculated using Equation (13).

$$D_{MC}(F||G) = \frac{1}{n}\sum_{i=1}^{n} log\ F(x_i)/G(x_i) \rightarrow D(F||G) \tag{13}$$

2.4. Forged Color Image Detection Algorithm Based on Similarity Measurement of GMM

The proposed algorithm for detection forged images generated by deep learning is as follows:

Algorithm 2: Forgery detection algorithm

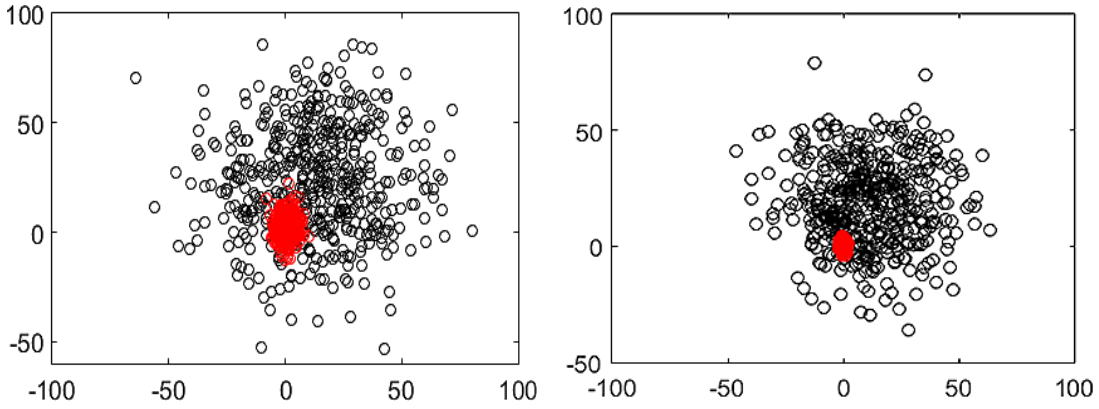
Input: feature matrixes of 500 real images in training dataset, feature matrixes of 500 forged images in training dataset, an arbitrary image in testing dataset
Output: testing result of identification
Step1: input the feature matrixes extracted from 500 real images to fit a GMM (F1)
Step2: input the feature matrixes extracted from 500 forged images to fit a GMM (F2)
Step3: input the feature matrix of a testing image and take the parameters obtained by F1 as a priori to fit a new

GMM (G1).
Step4: input the feature matrix of the same testing image as in step3, and take the parameters obtained by F2 as a priori to fit a new GMM (G2).
Step5: calculate the similarity between F1 and G1 (Dmc1) and that between F2 and G2 (Dmc2) respectively with the Monte Carlo Simulation.
Step6: if Dmc1 is less than Dmc2, the testing image input is a real image, otherwise not.

This method constructs feature matrixes with the value of (a, b) for every pixel in Lab space from the training dataset of real images and forged images respectively to fit GMM. In order to identify an image, the feature of images in the real dataset should be different from that in the forged image dataset as far as possible. If not, it is hard for the computer to identify the testing image and classify it.

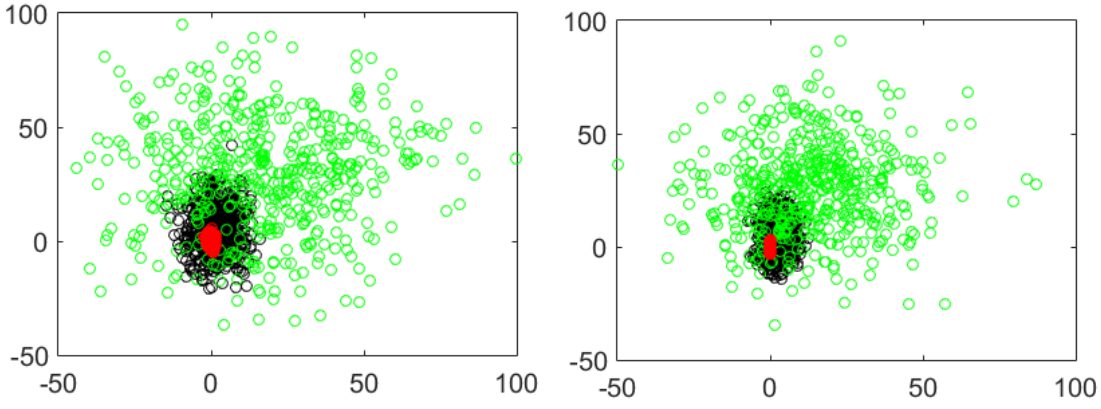
Therefore, firstly, the color feature of images in real datasets and forged datasets are taken as data points to fit GMM respectively in case of different cluster, and are displayed in two-dimensional space, as shown in Figure 1, Figure 2 and Figure 3. When the cluster is equal to 2, there is little difference between the two GMMs in two-dimensional space. When the cluster equals 5, the difference reaches the maximum.

Secondly, the two groups of GMM parameters obtained are taken as priori to fit a new GMM with the EM adaptation algorithm for each testing image. A real image and a forged image are taken randomly from the testing dataset to fit a new GMM with EM adaptation algorithm respectively, and then displayed in two-dimensional space as shown in Figure 4 and Figure 5.



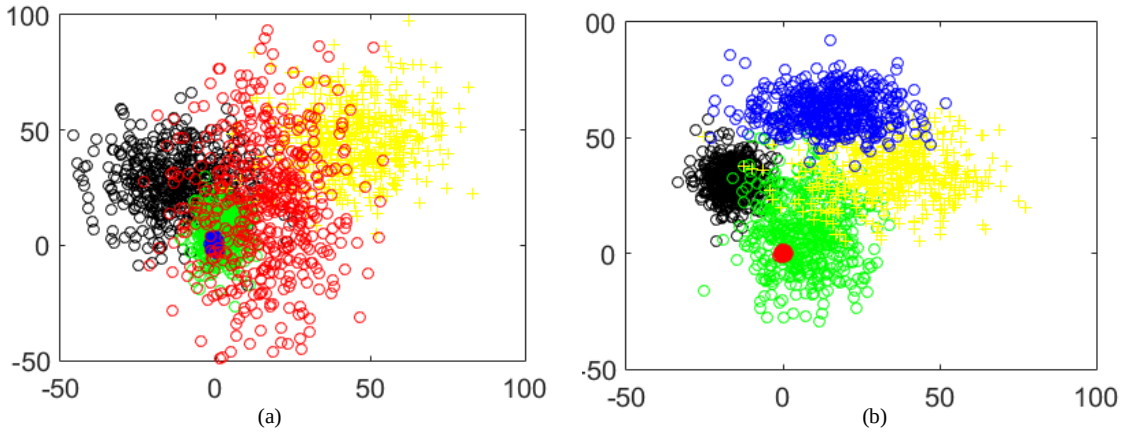
(a) (b)

Figure 1. (a) is the GMM of the 500 image color features in the real dataset when cluster is equal to 2, (b) is the GMM of the 500 image color features in the forgery dataset when cluster is equal to 2



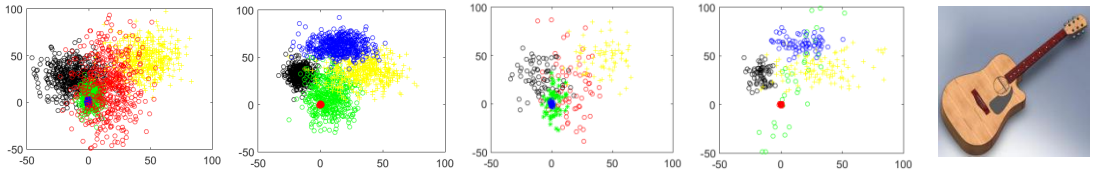
(a) (b)

Figure 2. (a) is the GMM of the 500 image color features in the real dataset when cluster is equal to 3, (b) is the GMM of the 500 image color features in the forgery dataset when cluster is equal to 3



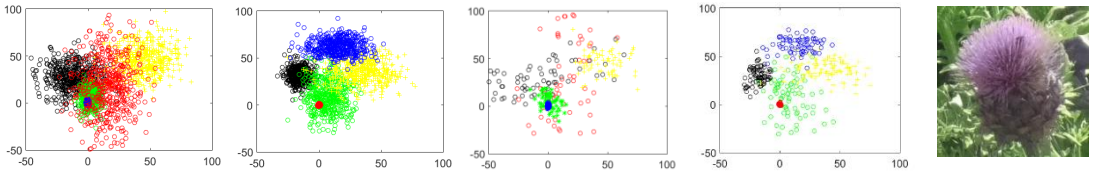
(a) (b)

Figure 3. (a) is the GMM of the 500 image color features in the real dataset when cluster is equal to 5, (b) is the GMM of the 500 image color features in the forgery dataset when cluster is equal to 5



(a) (b) (c) (d) (e)

Figure 4. (a) and (b) are the GMM of the 500 image color features in the real and forged dataset respectively, (c) and (d) are the GMM of the testing image (e) chosen randomly from the training set which is fit with parameters obtained from GMM of 500 images in the real dataset and forged dataset respectively as priori



(a) (b) (c) (d) (e)

Figure 5. (a) and (b) are the GMM of the 500 image color features in the real and forged dataset respectively, (c) and (d) are the GMM of the testing image (e) chosen randomly from the training set which are fit with parameters obtained from GMM of 500 images in the real dataset and forged dataset respectively as priori

Finally, the similarity between two GMMs is calculated to identify the testing image. Generally, Kullback-Leibler Divergence (KLD) [9] is adopted to measure the similarity. However, due to the absence of closed solution expression for KLD, Monte Carlo Simulation, which requires a large number of samples, is frequently adopted to calculate the similarity between two GMMs.

3. Experiment and discussion

The experiment dataset comes from the ImageNet [1]. 1,000 training sets are selected randomly and are recolored to construct our forged data set by applying Zhang R and other people's deep learning method. The colored effect is shown in Figure 6. The experiment is classified into three groups. The number of 250 natural images and 250 forged images are trained and tested in the first group. We calculate the accuracy of detection when the clusters take 2, 3, 4, 5, 6, respectively, before training the color features of natural images and forged ones of the training set directly through SVM, which is a traditional method. Then, the testing set is detected. The number of 500 natural images and 500 forged images are trained and tested in group two. We calculate the accuracy of detection when the clusters take 2, 3, 4, 5, 6, respectively, before training the color features of natural images and forged ones of the training set directly through the traditional SVM. Then, the testing set is detected. While in group three, 500 natural images and 500 forged images are halved in their sizes and trained afterwards, which is followed by the accuracy calculation of detection when the clusters take 2, 3, 4, 5, 6, respectively, before training the color features of natural images and forged ones of the training set directly through the traditional SVM.

Figure 6. The first line is the image of a part of the real datasets, and the second line is a part of forged image datasets

We set three groups of experiments and utilize our method for verification. Because our experimental data sets are randomly selected in ImageNet datasets, the robustness of our algorithm is guaranteed. Through experiments, we found that the experimental result that is affected by many factors indicates that different number of clusters affect the accuracy of detection severely. Based on the experiment, we found that the detection accuracy is the highest when the cluster takes 5. The excessive or few number of images in the training set causes the cases of fitting and under-fitting in the process of SVM training, affecting the accuracy of the experimental result deeply. In addition, we also find that by resizing the length and width of the datasets to 1/2, some key features of the image will disappear and the experimental results will become worse. The concrete results are demonstrated in Table 1.

Table 1. Detection accuracy for three groups of image sets. Group1:250 natural images and 250 forged images; Group 2: 500 natural images and 500 forged images; Group 3: 500 natural images and 250 forged images (the length and width of the original 1/2)

	Group 1:	Group 2:	Group 3:
The method we propose (cluster = 2)	52.7%	53.4%	52.3%
The method we propose (cluster = 3)	54%	61.3%	53.7%
The method we propose (cluster = 4)	55.7%	60%	55%
The method we propose (cluster = 5)	62.3%	67%	63.7%
The method we propose (cluster = 6)	56.5%	59%	58.4%
The method of SVM	54.4%	55.3%	52.2%

4. Conclusions

There is still no detection algorithm for forged images generated by deep learning. This paper aims to calculate the similarity between Gaussian Mixture Distributions by the fitting of GMM and EM adaptation algorithm. Due to the absence of closed solution for the distance between Gaussian Mixture Distributions, Monte Carlo Simulation has been adopted to calculate it. The experimental results demonstrate that the proposed method is quite more accurate than the traditional choice of SVM. For further study, more low-level feature of images will be considered, and the feature expression of the image will be enhanced to improve the accuracy of detection.

Acknowledgements

The research is supported by the National Nature Science Fund Projects (61562093, 61661051) and the Key Project of Applied Basic Research Program of Yunnan Province (2016FA024).

References

1. J. Deng, W. Dong, R Socher, et al, “ImageNet: a Large-Scale Hierarchical Image Database” in Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009:248-255.
2. H. Farid, S. Lab, “Creating and detecting doctored and virtual images: Implications to the Child Pornography Prevention Act [J]. 2004.
3. J. R. Hershey, P. A. Olsen, “Approximating the Kullback-Leibler Divergence Between Gaussian Mixture Models” in IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2007: IV-317 - IV-320.
4. W.R. Howard, “Pattern Recognition and Machine Learning”, [M]. Springer, 2006.
5. E. Luo, S. H. Chan, T. Q. Nguyen, “Adaptive Image Denoising by Mixture Adaptation” in IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society, 2016, 25(10):4489-4503.
6. R. L. Rivest, A. Shamir, L. Adleman, “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems” in Communications of the ACM, 1983, 26(2):96-99.

7. P. Sutthiwan, Y. Q. Shi, J. Dong, et al, “New Developments in Colour Image Tampering Detection” in IEEE International Symposium on Circuits and Systems. IEEE, 2010:3064-3067.
8. V. Thirunavukkarasu, J. S. Kumar, “Passive Image Tamper Detection Based on Fast Retina Key Point Descriptor” in IEEE International Conference on Advances in Computer Applications. IEEE, 2017:279-285.
9. N. Vasconcelos, P. Ho, P. Moreno, “The Kullback-Leibler Kernel as a Framework for Discriminant and Localized Representations for Visual Recognition” in Computer Vision - ECCV 2004. DBLP, 2004:430-441.
10. Y. E. Xi, “Blind Image Tamper Detection Algorithm Based on Radon and Fourier-Mellin Transform” in Signal Processing, 2010:212-215.
11. J. M. Yang, T. Q. Huang, W. J. Jiang, “Splicing Image Tamper Detection Based on Human Face Colour Temperature” in Journal of Shandong University, 2013.
12. J. Zhong, Y. Gan, J. Young, et al, “Copy Move Forgery Image Detection via Discrete Radon and Polar Complex Exponential Transform-Based Moment Invariant Features”, in International Journal of Pattern Recognition and Artificial Intelligence, 2017, 31(02): 1754005.
13. R. Zhang, P. Isola, A. A. Efros, “Colourful Image Colorization” in European Conference on Computer Vision. Springer, Cham, 2016:649-666.