A statistical feature based approach to distinguish PRCG from photographs

Xiaofeng Wang\(^a\), Yong Liu\(^a\), Bingchao Xu\(^a\), Lu Li\(^b\), Jianru Xue\(^b\)

\(^a\) School of Science and Shaanxi Key Laboratory for Network Computing and Security Technology, Xi’an University of Technology, Xi’an, Shaanxi 710048, PR China
\(^b\) Institute of Artificial Intelligence and Robotics at Xi’an Jiaotong University, Xi’an, Shaanxi 710049, PR China

**ABSTRACT**

We present a passive forensics method to distinguish photorealistic computer graphics (PRCG) from natural images (photographs). The goals of our work are to improve the detection accuracy and the robustness to content-preserving image manipulations. In the proposed method, Homomorphic filtering is used to highlight the detail information of image. We find that the texture changes are different between photographs and PRCG images under same Homomorphic filtering transformation, and then we use the difference matrices to describe the differences of texture changes. We define a customized statistical feature, named texture similarity, and combine it with the statistical features extracted from the co-occurrence matrices of differential matrices to construct forensics features. Then we develop a statistical model and use SVM as classifier to distinguish PRCG from photographs. Experimental results show that the proposed method enjoys following advantages: (1) Proposed method reaches higher detection accuracy, synchronously, it is robust to tolerate content-preserving manipulations such as JPEG compression, adding noise, histogram equalization, and filtering. (2) Proposed method is provided with satisfactory generalization capability, it will be available when the training samples and the testing samples come from different sources.

© 2014 Elsevier Inc. All rights reserved.

**1. Introduction**

With the rapid growth and widespread using of digital cameras and mobile phones, photos and pictures can be obtained when and where. At the same time, rapid development of computer vision and computer graphic technologies, make it is more convenient to generate the photorealistic computer graphics (PRCG). These PRCG are so realistic that it is very difficult to differentiate them from photographs and pictures by human eyes. See Fig. 1. The old adage of “seeing is believing” is no longer true. This trend has brought new issues and challenges for us concerning the authenticity of digital images. Therefore, identifying PRCG from photographs has become an important topic in the field of media application.

Photographs are those pictures that are captured from real-world by imaging device such as camera. PRCG are scene images that does not exist in reality; it is generated only by computers and graphic software through the following process: first, a 3D polygon model is built to simulate a desire shape, then color, texture and simulate light irradiation are given to this model. At last, PRCG is imaged using a virtual camera. Recent advances in computer graphics are bring an amazing result, that is, it is possible to create models that are physically accurate without visual artifacts and lighting inconsistencies. Therefore, distinguishing PRCG from photographs is a very challenging problem.

There are many established methodologies to distinguish PRCG from photographs in recent years. The majority of existing methods in literature are based on statistical learning model, in which, the features of inconsistency between photographs and PRCG are used for classification. Therefore, the main difference of existing approaches lies on the features extraction, and features extraction methods can be classified into two categories, the first is based on transform domain and the second is based on the physical characteristics of the imaging equipment.

Taking a general survey on existing methods, there are some problems such as lower detection precision, weaker robustness and unsatisfactory stability. In our work, we aim at improving the detection accuracy, robustness, and generalization capability, and seek for new features that are able to distinguish photographs and PRCG. We find that the texture changes are different between
features were generated by using the statistical moments of image. A similar method was presented by Chen et al. [5], in which, 234-D features were extracted from the wavelet coefficient histogram. The method was proposed by Wang and Moulin [4] where 144-D features were generated by using the first 4-order statistics (mean, variance, skewness, and kurtosis) of the sub-band coefficients and the statistical features extracted from the co-occurrence matrix of differential matrixes to construct forensics features. Then we develop a statistical model, and use the least squares SVM as a classifier to classify PRCG from photographs. Experimental results show that the proposed method not only possesses satisfactory detection accuracy, also, it is robust to tolerate content-preserving manipulations, and provides satisfactory generalization capability for different image data sets.

The rest of this paper is organized as follows. In Section 2, we introduce the relational works. The proposed method is described in Section 3. Section 4 contains experimental results. Finally, a short summary are presented in Section 5.

2. Related works

The earliest approaches to distinguish photographs and PRCG were proposed by Ng and Chang [1] and Lyu and Farid [2], respectively. Work [1] presented a classification approach that used the natural image statistics features composed of 33-dimensional (33-D) power spectrum features, 24-D local patch features and 72-D wavelet higher-order statistics. Subsequently, a geometry-based approach [3] was proposed by the same authors to model the physical differences between PRCG and photographs, in which, 192-D features were extracted through analyzing local patch statistics, local fractal dimension, surface gradient, quadratic geometry, and Beltrami flow. Lyu and Farid [2] developed a statistical model based on a wavelet-like decomposition, in which, 216-D features were generate by using the first 4-order statistics (mean, variance, skewness, and kurtosis) of the sub-band coefficients and the inter-sub-band prediction errors. Another wavelet transform based method was proposed by Wang and Moulin [4] where 144-D features were extracted from the wavelet coefficient histogram. The similar method was presented by Chen et al. [5], in which, 234-D features were generated by using the statistical moments of image characteristic functions and the wavelet sub-band coefficients in HSV color space. Cui et al. [6] used a similar method, in which, three frequency-domain moments of the wavelet coefficient histogram were extracted, and obtained 78-D features were used as classification features. The detection rate of this method is 94%. Zheng and Ping [7] used 81-D features generated by adjacent pixels coherence histogram and 120-D co-occurrence matrix features extracted from HSV color space as distinguishing features to identify PRCG from photographs. In Sankar et al. [8], authors proposed an approach that based on an aggregate of existing features and texture interpolation features to distinguish PRCG from real images. In their work, 80-D features were generated, in which, 44-D came from the moment-based method, 24-D came from the texture interpolation method, 6-D came from color histogram, and 6-D came from patch statistics. Li et al. [9] proposed a distinguishing method, in which, the 2-order difference statistics of the images and the first 4-order statistics of its predicting error images were extracted to form 144-D distinguishing features.

Since the majority of real images are obtained by imaging equipments, some physical characteristics of imaging equipments are always remained in imaging process, and these characteristics can be used to distinguish digital photographs from PRCG. Dehnie et al. [10] proposed an approach that used different pattern noise between photographs and PRCG as distinguishing features. Dirik et al. [11] identified photographic images by detecting Bayer color filter array (CFA) and the traces of chromatic aberration. Considering that most digital cameras employ the image sensor with a color filter array, the process of demosaicing interpolates the raw image to produce at each pixel an estimate for each color channel, Gallagher and Chen presented method [12] to distinguish PRCG from photographs by detecting the interpolation traces caused by the Bayer color filter array. This algorithm reaches the detection accuracy of 98.4% in Columbia open data set. It is almost the highest among similar algorithms in the same dataset. But the algorithm is less robust to JPEG compression and noise because such manipulations may destroy the interpolation traces caused by the Bayer color filter array. The other high accuracy method was proposed by Li et al. [13], it was reported that the detection accuracy can reach as high as 98.3% for distinguishing Computer Graphics (CG) from photographic images in their own data set. In their data set, computer graphics are composed of [14,15], and photographic images are collected by authors. Zhang et al. [16] proposed a method that distinguishes PRCG from photographic images using visual characteristics.
vocabulary on local image edges. They preprocessed image edge patches and projected it into a 7-D sphere, and then a visual vocabulary was constructed via determining the key sampling points in accordance with Voronoi cells. Recently, Farid and Bravo [17], Dang-Nguyen et al. [18,19] presented methods that can discriminate computer generated and natural human faces. Literature [20] presented more technical approaches, system design, and other practical issues in tackling the distinction between photographic images and PRCG.

3. Proposed method

Although PRCG are so realistic that it is very difficult to be differentiated from photographs by human’s eyes, but the imaging mechanism and imaging process are completely dissimilar. The imaging process of the photographs is very complex since it is inevitably influenced by both natural environment and acquisition equipment, while the process of generating a PRCG only primitively simulates the imaging process of photographs. These differences will lead to different characteristics in many aspects, especially, in texture detail. Generally speaking, the texture details of photographs appear to smooth while those of PRCG appear to quavering [9]. Additional, photographs are always influenced by natural environment. If illumination is uneven, the average brightness of the various parts on image would shudder, and image detail structure that corresponding to dark regions would difficult to be recognized. All these differences can be used to distinguish PRCG from photographs. To this end, we keep finding the ways to capture and describe the essential differences between photographs and PRCG.

In digital image processing, the approach of Homomorphic filtering is derived from an illumination reflectance model of the image, and it can be used to perform simultaneous dynamic range compression and contrast enhancement to image. For photographs, after Homomorphic filtering processing, the intensity variation would be reduced while the details would be highlighted. However, such a variation is not obvious for PRCG after the same Homomorphic filtering processing. The reason is that the natural texture expression is different from that of the PRCG. See Fig. 2. In Fig. 2(a) and (b) are photographs, and (c) and (d) are PRCG. We can see that after Homomorphic filtering, the texture details of photographs in shaded areas are apparent, but the PRCG has not such a change. For this reason, we can use Homomorphic filter to capture the detail difference between photographs and PRCG.

Additional, Contourlet transformation [21,22] is able to capture a sparse expansion for typical images that are piecewise smooth away from smooth contours. The main difference between Contourlet transform and wavelet transform is that the latter cannot capture the geometrical smoothness of the contours. Motivated by this fact, the statistics of the Contourlet sub-bands are used to construct distinguishing features. The least squares SVM are used as classifier for classifying. Our method can be divided into 2 stages: feature extraction and classification. The framework is shown in Fig. 3.

3.1. Feature extraction

For $N \times N$ image $I$, taking 2-level Contourlet decomposition, the image is decomposed into low-frequency sub-band and band-pass directional sub-bands. See Fig. 4.

We denote the coefficient matrixes as $M$, $M_1$, and $M_2$, respectively, and corresponding sub-band images are noted in Fig. 4.

\[
M = \begin{bmatrix}
m_{11} & m_{12} & \cdots & m_{i1} \\
m_{21} & m_{22} & \cdots & m_{ij} \\
\vdots & \vdots & \ddots & \vdots \\
m_{i1} & m_{i2} & \cdots & m_{ij}
\end{bmatrix}
\] (1)

Fig. 2. The visual effects of photograph and PRCG before and after Homomorphic filtering.
Let \( \text{Hom}(\cdot) \) represents a convolution Homomorphic filtering transformation, and let \( I = \text{Hom}(I) \). \( I \) and \( I' \) can be denoted as:

\[
M_s^m = \begin{bmatrix}
m_{11}^m & m_{12}^m & \cdots & m_{1q}^m \\
m_{21}^m & m_{22}^m & \cdots & m_{2q}^m \\
\vdots & \vdots & \ddots & \vdots \\
m_{p1}^m & m_{p2}^m & \cdots & m_{pq}^m 
\end{bmatrix}, \quad \text{where } s = 1, 2, 3, 4 \quad (2)
\]

\[
M_t^h = \begin{bmatrix}
m_{11}^h & m_{12}^h & \cdots & m_{1y}^h \\
m_{21}^h & m_{22}^h & \cdots & m_{2y}^h \\
\vdots & \vdots & \ddots & \vdots \\
m_{x1}^h & m_{x2}^h & \cdots & m_{xy}^h 
\end{bmatrix}, \quad \text{where } t = 1, 2, 3, \ldots, 8 \quad (3)
\]

\[
I = \begin{bmatrix}
a_{11} & a_{12} & \cdots & a_{1n} \\
a_{21} & a_{22} & \cdots & a_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{n1} & a_{n2} & \cdots & a_{nn} 
\end{bmatrix} \quad (4)
\]

\[
I' = \begin{bmatrix}
a'_{11} & a'_{12} & \cdots & a'_{1n} \\
a'_{21} & a'_{22} & \cdots & a'_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
a'_{n1} & a'_{n2} & \cdots & a'_{nn} 
\end{bmatrix} \quad (5)
\]

We take same 2-level Contourlet decomposition for the image \( I \), and let \( M_s^m \), \( M_t^h \) and \( M_s^h \) represent the coefficient matrices, respectively.
In order to represent the difference of the texture detail between photographs and PRCG, we define 4 categories difference matrices $D_1^i$, $D_1^{m}$ and $D_2^i$: 

$$D_1^i = I - I'$$  

$$D_1^{m} = M - M^d$$  

$$D_2^m = M_1 - M_2$$, where $s = 1, 2, 3, 4$  

$$D_2^t = M_1 - M_2$$, where $t = 1, 2, 3, \ldots, 8$  

Considering that the Gray Level Co-occurrence Matrix (GLCM) is referred to as a co-occurrence distribution of the image pixels, the distribution of the co-occuring differences is defined over an image $I$ and the distribution of the co-occuring differences is defined over an image $I'$. 

In order to represent the difference of the texture detail between photographs and PRCG, we define 4 categories difference matrices $D_1^i$, $D_1^{m}$ and $D_2^i$: 

$$g(i, j) = \sum_{p=1}^{N} \sum_{q=1}^{M} \begin{cases} 1, & \text{if } (p, q) = (i, j) \\
0, & \text{otherwise} \end{cases}$$

where $i$ and $j$ are the intensity values of the image $I$, $p$ and $q$ are the spatial positions in image $I$, and the offset $(\Delta x, \Delta y)$ depends on the direction at which the matrix is computed.

In our scheme, the parameters of gray-level co-occurrence matrices are chosen as follows. Assuming the gray-levels of the gray image is from 0 to $L-1$, the gray-level co-occurrence matrix offset parameter $\Delta x$ is set to 0, and $\Delta y$ is set to 0, namely, $\theta = 0$, this means that only the nearest neighbors in horizontal direction are considered for gray-level co-occurrence matrix calculation. For 13 difference matrices $D_1^i$, $D_1^{m}$, $D_2^m$ ($s = 1, 2, 3, 4$) and $D_2^t$ ($t = 1, 2, 3, \ldots, 8$), their co-occurrence matrices $G_k$ ($k = 0, 1, 2, \ldots, 13$) can be denoted as follows, respectively.

$$G_k = \begin{bmatrix} g_k(0, 0) & \cdots & g_k(0, L - 1) \\
\vdots & \ddots & \vdots \\
g_k(L - 1, 0) & \cdots & g_k(L - 1, L - 1) \end{bmatrix}$$

When $k = 0$, $G_k$ represents the GLCM of the difference matrices $D_1^i - I'$. 

Co-occurrence matrices capture the features of the texture but they are not directly useful for further analysis, such as the comparison of two image textures. In order to represent the texture details more compactly, we estimate the following five kinds of numeric features (statistics) of each above mentioned co-occurrence matrix, respectively.

1. Energy. It is the quadratic sum of elements in GLCM.

$$f_1^i = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} (g_k(i, j))^2$$

2. Contrast. It is a measure of the intensity contrast between a pixel and its neighbor pixels over the whole image.

$$f_2^i = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} (i - j)^2 g_k(i, j)$$

3. Homogeneity. It measures the tightness of elements distribution to diagonal in GLCM.

$$f_3^i = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \frac{g_k(i, j)}{1 + |i - j|}$$

4. The maximum of column mean.

$$R_k = \sum_{i=0}^{L-1} g_k(i, j), \quad j = 1, 2, 3, \ldots, L$$

5. Texture similarity.

In order to quantitatively measure the texture change before and after Homomorphic filtering, we define a term named “Texture similarity”. Then we compute the texture similarity of difference matrices $D_1^i$, $D_1^{m}$, $D_2^m$ and $D_2^t$. For simplicity, we denote the difference matrices uniformly over them, and denote as $D_k$, $k = 0, 1, 2, 3, \ldots, 13$. Then we estimate the texture similarity for each difference matrix, respectively.

For the difference matrices $D_k$ ($k = 0, 1, 2, 3, \ldots, 13$):

$$D_k = \begin{bmatrix} d_{11} & \cdots & d_{1n} \\
\vdots & \ddots & \vdots \\
d_{m1} & \cdots & d_{mn} \end{bmatrix} = \begin{bmatrix} A_{k1}^T & \cdots & A_{kj}^T & \cdots & A_{kn}^T \end{bmatrix}$$

Here, $A_{ki} = \{d_{1i}, d_{2i}, \ldots, d_{mi}\}$.

We define an equivalence relation $R$ on the set: $A_{ki}$

$$d_{ki}^R d_{kj}^R \iff (d_{ki}^R, d_{kj}^R) \in A_{ki}$$

It means: $d_{ki}^R$ and $d_{kj}^R$ satisfy the equivalence relation $R$, if and only if the following two conditions must be satisfied:

$$d_{ki}^R = d_{kj}^R \quad \text{and} \quad d_{ki}^R, d_{kj}^R \in A_{ki}$$

Symbol “$\iff$” represent “if and only if” or “be equivalent to”, “$\iff$” represent “and”.

Then the equivalence class of each element in $A_{ki}$ can be denoted as:

$$\forall d_{ki}^R \in A_{ki}, \quad \{d_{kj}^R \mid (d_{kj}^R \in A_{ki}) \wedge (d_{ki}^R R d_{kj}^R)\}$$

Let $E_j^R = \arg \max_{d_{kj}^R} \|d_{kj}^R\|$. 


Here, \( |\cdot| \) denote the operator to compute the cardinality of a set.

Let \( f_i^k = \sum_{j=1}^{N} E_{ij} \) \hspace{1cm} (24)

We named \( f_i^k \) as “Texture similarity”.

Then 70-dimension eigenvectors are defined as follows:

\[
V = \{f_0^1, f_0^2, \ldots, f_0^6, f_1^1, f_1^2, \ldots, f_1^{13}, f_1^{14}, \ldots, f_5^1, f_5^2, \ldots, f_5^{13}, f_5^{14}\} \hspace{1cm} (25)
\]

3.2. Classification

Support Vector Machine (SVM) [23] is a state-of-the-art classification method that based on the statistical learning theory, and has been applied successfully in text classification, image recognition and biological information processing. In our work, we take feature vector \( V \) as distinguishing features, and the LS-SVM classifier is used to distinguish photograph and PRCG, and radial basis function (RBF) is used as the kernel function.

To train classifier, we labeled the PRCG as ‘‘-1’’ and the photograph as ‘‘1’’, used the feature vector \( V \) as classification feature, then took all of them as the input of the LS-SVM to obtain classifier through learning, this process can be described in Fig. 5.

4. Experimental results and analysis

In this section, we use experimental results to demonstrate the performance of the proposed method. Our algorithm is implemented and tested using MATLAB2010a, and our experiment is performed on the computer with Processor Pentium(R) Dual-Core CPU, E5400 @ 2.70 GHz, 2.00 GB RAM.

4.1. Detection Accuracy and Performance Analysis

The goal of this experiment is to test the detection accuracy and analyze the performance of the proposed algorithm. To this end, we present the formalization definitions of true negative rate and true positive rate as follows:

\[
P_{TN} = \text{True negative rate} = \frac{\text{The number of natural image (CG) is detected as natural image (CG)}}{||\text{Testing sample set}||} \times 100\% \hspace{1cm} (26)
\]

\[
P_{TP} = \text{True positive rate} = \frac{\text{The number of natural image (CG) is detected as CG (natural image)}}{||\text{Testing sample set}||} \times 100\% \hspace{1cm} (27)
\]

The true negative rate represents the detection accuracy of the algorithm, it means the ratio that photographs are detected as photographs, and PRCG are detected as PRCG. In order to test this index, we performed our experiments on Columbia University Image Database [24]. Tested images contained a variety of images including various brightness, textures and fine details. We take randomly 400 photographs and 400 PRCG as training samples, and take randomly other 100 photographs and 100 PRCG as testing samples, use LS-SVM as classifier.

Let \( P_{TN(PCG)} \) represents the detection rate that testing samples are photographs and PRCG, \( P_{TN(PG)} \) represents the detection rate that testing samples are photographs, \( P_{TP(PCG)} \) represents the detection rate that testing samples are PRCG.

Table 1 presents 10 detection results and comparing results with [6,12], in each testing, the training samples and the testing samples are randomly selected. As can be seen from Table 1, the true negative rate of proposed method is satisfactory whatever the testing samples contain either photographs or PRCG images.

As can be seen from Table 1, proposed method is superior to the method [6] in term of detection rate, and superior to the method [12] when testing samples only contain photographs, testing samples contain photographs and PRCG, and less than the method [12] when testing samples only contain PRCG.

In order to confirm the availability of the proposed method, we tested 20 pair true negative rate and true positive rate of the proposed method, as well as [6,12]. In experiment, we generated training samples by taking randomly 400 photographs and 400 PRCG images from Columbia Image Database, and generated testing samples by taking other 10n photographs and 10n PRCG images, respectively, here, \( n = 1, \ldots, 20 \), taking LS-SVM as classifier to get \( P_{TN} \) and \( P_{TP} \). The results are shown in Table 2.

Literature [20] lists the classification performance of the various methods on different data sets. We compare some results in Columbia open data set, the results are listed in Table 3.

There are other methods such as [18,19] are provided with high classification accuracy, but they are distinguishing between computer generated and natural faces based on facial expressions. Due to aim at face image, it is not appropriate to test the accuracy of [18,19] in our data set. Work [13] was reported that the detection accuracy can reach as high as 98.3% for distinguishing Computer Graphics (CG) from photographic images in their own data set. In their data set, computer graphics are composed of [14,15], and photographic images are collected by authors. However, in our experiments, the highest achievable accuracy is 87.55% by performing their code in our data sets. We further investigated the feature used in [13]. Work [13] is a method that uses the statistical difference of uniform gray-scale invariant local binary patterns (LBP) to distinguish PRCG from photographs. In [13], the original JPEG coefficients of Y and Cr components, and their prediction errors were used for LBP calculation, and 236-D LBP features were obtained and SVM was used for classification. LBP is a kind of

![Fig. 5. Training and testing process of LS-SVM classifier.](image-url)
is very common occurrences. E.g., work [25] is reported that the
image resolutions of Columbia open data set are from 757 to 1152 × 768. All of these are possible reasons to lead to declining
performance of the algorithm. We will further discuss in Section 4.3.

Table 1
Detection rates and comparison results.

<table>
<thead>
<tr>
<th>Experiment index</th>
<th>Proposed method</th>
<th>Cui et al. [6]</th>
<th>Gallagher and Chen [12]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PTP (%)</td>
<td>PFP (%)</td>
<td>PTP (%)</td>
</tr>
<tr>
<td>1</td>
<td>96.5</td>
<td>94.5</td>
<td>97.5</td>
</tr>
<tr>
<td>2</td>
<td>97.5</td>
<td>95.5</td>
<td>96.5</td>
</tr>
<tr>
<td>3</td>
<td>98.0</td>
<td>96.5</td>
<td>96.0</td>
</tr>
<tr>
<td>4</td>
<td>95.5</td>
<td>95.0</td>
<td>96.5</td>
</tr>
<tr>
<td>5</td>
<td>97.5</td>
<td>97.5</td>
<td>96.0</td>
</tr>
<tr>
<td>6</td>
<td>96.0</td>
<td>96.5</td>
<td>97.0</td>
</tr>
<tr>
<td>7</td>
<td>96.5</td>
<td>95.0</td>
<td>95.5</td>
</tr>
<tr>
<td>8</td>
<td>97.5</td>
<td>96.5</td>
<td>97.5</td>
</tr>
<tr>
<td>9</td>
<td>95.5</td>
<td>95.0</td>
<td>96.0</td>
</tr>
<tr>
<td>10</td>
<td>98.0</td>
<td>96.5</td>
<td>98.0</td>
</tr>
</tbody>
</table>

Table 2
P_TN and P_TP and comparison results.

<table>
<thead>
<tr>
<th>n</th>
<th>Testing samples are photographs</th>
<th>Testing samples are PRRC images</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P_TN (%)</td>
<td>P_TP (%)</td>
</tr>
<tr>
<td>1</td>
<td>100.00</td>
<td>0.00</td>
</tr>
<tr>
<td>2</td>
<td>100.00</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>100.00</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>100.00</td>
<td>0.00</td>
</tr>
<tr>
<td>5</td>
<td>100.00</td>
<td>0.00</td>
</tr>
<tr>
<td>6</td>
<td>98.67</td>
<td>3.33</td>
</tr>
<tr>
<td>7</td>
<td>97.14</td>
<td>2.86</td>
</tr>
<tr>
<td>8</td>
<td>96.67</td>
<td>3.33</td>
</tr>
<tr>
<td>9</td>
<td>97.78</td>
<td>2.22</td>
</tr>
<tr>
<td>10</td>
<td>98.00</td>
<td>2.00</td>
</tr>
<tr>
<td>11</td>
<td>97.27</td>
<td>2.73</td>
</tr>
<tr>
<td>12</td>
<td>97.50</td>
<td>2.50</td>
</tr>
<tr>
<td>13</td>
<td>99.23</td>
<td>0.71</td>
</tr>
<tr>
<td>14</td>
<td>99.29</td>
<td>0.71</td>
</tr>
<tr>
<td>15</td>
<td>98.00</td>
<td>2.00</td>
</tr>
<tr>
<td>16</td>
<td>99.38</td>
<td>0.62</td>
</tr>
<tr>
<td>17</td>
<td>97.06</td>
<td>2.94</td>
</tr>
<tr>
<td>18</td>
<td>98.89</td>
<td>1.11</td>
</tr>
<tr>
<td>19</td>
<td>97.89</td>
<td>2.11</td>
</tr>
<tr>
<td>20</td>
<td>98.00</td>
<td>2.00</td>
</tr>
</tbody>
</table>

Table 3
Classification accuracy comparison.

<table>
<thead>
<tr>
<th>Work</th>
<th>The approach of feature extraction</th>
<th>Feature dimension</th>
<th>Data set</th>
<th>Highest classification accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ng et al. [3]</td>
<td>Fractal geometry, differential geometry</td>
<td>192</td>
<td>Columbia open data set</td>
<td>83.5</td>
</tr>
<tr>
<td>Chen et al. [5]</td>
<td>Statistical moments, wavelet coefficients</td>
<td>234</td>
<td>Columbia open data set</td>
<td>82.1</td>
</tr>
<tr>
<td>Cui et al. [6]</td>
<td>Statistical moments, wavelet coefficients</td>
<td>78</td>
<td>Columbia open data set</td>
<td>94.0</td>
</tr>
<tr>
<td>Sankar et al. [8]</td>
<td>Combining features</td>
<td>557</td>
<td>Columbia open data set</td>
<td>98.0</td>
</tr>
<tr>
<td>Gallagher and Chen [12]</td>
<td>Interpolation clues related to the Bayer color filter array</td>
<td>1</td>
<td>Columbia open data set</td>
<td>98.4</td>
</tr>
<tr>
<td>Proposed method</td>
<td>Statistical feature</td>
<td>70</td>
<td>Columbia open data set</td>
<td>98.0</td>
</tr>
</tbody>
</table>

typical texture feature, it strongly depends on the image texture characters and image resolution. Due to work [13] only use LBP feature directly, it is inevitable that the detection results are closely related to used images data set. Moreover, the image resolution used in [13] is restricted from 500 × 500 to 800 × 800, while the image resolutions of Columbia open data set are from 757 × 568 to 1152 × 768. All of these are possible reasons to lead to declining accuracy when perform method [13] in our data set. Such situation is very common occurrences. E.g., work [25] is reported that the accuracy reaches 99.33%. However, work [13] reported that the accuracy decline to 69.94% when it was tested in the data set of [13]. Keeping steady accuracy in different database is a good performance of the algorithm. We will further discuss in Section 4.3.

To illustrate the superiority of our method in a more convincing way, we only present the comparison results of our algorithm and some methods that are using Columbia open data set. We compare our method with [12] in detail, because method [12] reaches highest accuracy in Columbia open data set.

Receiver Operating Characteristic (ROC) curves is a most common performance evaluation method of the detection algorithms. However, the recently state-of-the-art [26] reported, in presence of composite hypotheses (due to unknown photographs and PRRC images), a ROC curve is not sufficient to sum up the performances of a detector. The reason is that ROC curves are especially relevant for testing two simple hypotheses. In the situation of discriminat-
positive rate, and compare it with some existing works. The results are shown in Fig. 6.

Fig. 6 shows the distribution and the comparison of true positive rates. We can see that the true positive rates of the proposed method are all no more than 5%, this means that the proposed method works well whatever the image number.

4.2. Robustness analysis

These experiments are performed to test the proposed method is robust for incidental changes caused by content-preserving manipulations, such as JPEG compression, adding noise, and Histogram equalization. Robustness is critical important because it determines the practical applicability of the algorithm. As the literature [20] notes “The quoted performance may not be a sufficient indicator for the usefulness of the methods when it comes to real applications.” Some methods work well on recognizing the original-size and uncompressed photographic images. However, the images extremely likely are manipulated, and undergo common image processing such as JPEG compression, adding noise, and so on in practical application. Therefore, our core work will be developing a robust distinguishing method that can tolerate common content-preserving manipulations.

The robustness can be expressed by the true negative rate in the instance that image have undergone content-preserving manipulations. Generally, we say an algorithm is robust to tolerate a certain kind of manipulation if it holds a satisfactory true negative rate after that manipulation. To test such a property, we generate training sample set using the similar method like Section 3.1. We generate testing sample sets by adding noise, JPEG compression, and Histogram equalization, respectively, for 200 randomly selected images from Columbia University Data Set. The results shown in Table 4 are actually the average values of the 10 testing, and training samples and testing samples were randomly selected in each testing.

As can be seen from Table 4, for content-preserving manipulations, our method presents satisfactory detection results. It means that our method is robust to tolerate content-preserving manipulations. The method proposed by Gallagher and Chen [12] works well on recognizing the original-size and uncompressed photographic images. But the algorithm is less robust to JPEG compression and noise because such manipulations may destroy the interpolation clues caused by the Bayer color filter array.

4.3. Generalization capability analysis

Generally, different image data sets have different influences to the detection results of the algorithm. In real world applications, the data sources are varied. It is impossible to train a classifier that is available for all images come from Internet. The generalization capability of the algorithm refers to the availability for different image databases. High generalization capability means that the

<table>
<thead>
<tr>
<th>Factor</th>
<th>Salt &amp; pepper noise (VAR = 0.001)</th>
<th>Histogram equalization</th>
<th>Mean filtering (3 x 3)</th>
<th>JPEG (QF = 30)</th>
<th>JPEG (QF = 60)</th>
<th>JPEG (QF = 90)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed method</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P_{TNP_{PG/CG}}$ (%)</td>
<td>90.00</td>
<td>81.50</td>
<td>85.00</td>
<td>87.50</td>
<td>86.00</td>
<td>94.50</td>
</tr>
<tr>
<td>$P_{TNP_{PG}}$ (%)</td>
<td>83.50</td>
<td>80.50</td>
<td>82.00</td>
<td>83.00</td>
<td>88.50</td>
<td>90.50</td>
</tr>
<tr>
<td>$P_{TNP_{CG}}$ (%)</td>
<td>90.50</td>
<td>70.50</td>
<td>86.00</td>
<td>86.50</td>
<td>84.00</td>
<td>93.50</td>
</tr>
<tr>
<td>Cui et al. [6]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P_{TNP_{PG/CG}}$ (%)</td>
<td>72.50</td>
<td>67.50</td>
<td>67.50</td>
<td>67.00</td>
<td>71.50</td>
<td>84.50</td>
</tr>
<tr>
<td>$P_{TNP_{PG}}$ (%)</td>
<td>94.00</td>
<td>56.00</td>
<td>90.00</td>
<td>80.50</td>
<td>92.50</td>
<td>90.50</td>
</tr>
<tr>
<td>$P_{TNP_{CG}}$ (%)</td>
<td>64.50</td>
<td>53.00</td>
<td>55.00</td>
<td>50.50</td>
<td>63.50</td>
<td>86.50</td>
</tr>
<tr>
<td>Gallagher and Chen [12]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P_{TNP_{PG/CG}}$ (%)</td>
<td>66.50</td>
<td>65.50</td>
<td>53.50</td>
<td>66.50</td>
<td>73.00</td>
<td>71.00</td>
</tr>
<tr>
<td>$P_{TNP_{PG}}$ (%)</td>
<td>88.00</td>
<td>88.00</td>
<td>37.00</td>
<td>40.00</td>
<td>41.00</td>
<td>46.50</td>
</tr>
<tr>
<td>$P_{TNP_{CG}}$ (%)</td>
<td>92.50</td>
<td>92.00</td>
<td>86.50</td>
<td>85.50</td>
<td>89.50</td>
<td>93.00</td>
</tr>
</tbody>
</table>
true negative rate of algorithm is satisfactory and move gradually within a small range when the training samples and testing samples come from different sources. In order to test the generalization capability of the proposed method, our experiments are performed on image sets from UCID [27], Columbia Image Database [24] and Art-CG gallery database [28]. Nine experiments are as follows:

1. Columbia → UCID: Training samples are generated by taking randomly 400 photographs and 400 PRCG images from Columbia Image Database, and testing samples are generated by taking randomly 200 photographs from UCID.
2. Columbia → UCID, Columbia: Training samples are generated by taking randomly 400 photographs and 400 PRCG images from Columbia Image Database, and testing samples are generated by taking 100 photographs from UCID and other 100 PRCG images from Columbia Image Database.
3. Columbia, UCID → Columbia: Training samples are generated by taking randomly 400 photographs from UCID, and 400 PRCG images from Columbia Image Database, and testing samples are generated by taking 100 photographs and 100 PRCG images from Columbia Image Database.
4. Columbia, UCID → Columbia: Training samples are generated by taking randomly 400 photographs from UCID, and 400 PRCG images from Columbia Image Database, and testing samples are generated by taking 200 photographs from Columbia Image Database.
5. Columbia, UCID → Columbia: Training samples are generated by taking randomly 400 photographs from UCID, and 400 PRCG images from Columbia Image Database, and testing samples are generated by taking other 200 PRCG images from Columbia Image Database.
6. Columbia → Art-CG gallery database: Training samples are generated by taking randomly 400 photographs and 400 PRCG images from Columbia Image Database, and testing samples are generated by taking randomly 200 PRCG images from Art-CG gallery database.
7. Art-CG gallery database, Columbia → Columbia: Training samples are generated by taking randomly 400 photographs from Columbia Image Database and 400 PRCG images from Art-CG gallery database, and testing samples are generated by taking randomly 100 photographs and 100 PRCG images from Columbia Image Database.
8. Art-CG gallery database, UCID → Columbia: Training samples are generated by taking randomly 400 photographs from UCID and 400 PRCG images from Art-CG gallery database, and testing samples are generated by taking randomly 100 photographs and 100 PRCG images from Columbia Image Database.
9. Art-CG gallery database, UCID → Columbia: Training samples are generated by taking randomly 400 photographs from UCID and 400 PRCG images from Art-CG gallery database, and testing samples are generated by taking randomly 200 PRCG images from Columbia Image Database.

Each experiment is repeated 10 times with different random images. Table 5 shows the statistical average of the true negative rate.

As can be seen from Table 5, for different data sets, the true negative rates of the proposed algorithm are satisfactory, and it varies in small range when the training samples and testing samples come from different sources. This shows that proposed method is available when the training samples and testing samples come from different sources. It means that the generalization capability of the proposed method is satisfactory.

5. Conclusions

Nowadays, computer imagery and computer generated images touch many aspects of our daily life. Computer graphics are getting so photorealistic that it has brought new challenges towards the credibility of digital images. Identifying photorealistic computer graphics from photographs has become an important topic in the field of media application.

In this paper, we aimed at improving the detection accuracy, robustness, and generalization capability, simultaneously, proposed a new approach to identify photorealistic computer graphics from photographic images. In the proposed method, the Homomorphic filter is used to highlight the image detail information, the statistical features of the Contourlet sub-bands of the image are used to construct the distinguishing features, and then the least squares SVM is used for classification. Experimental results show that the proposed method reaches high detection accuracy. Also, the algorithm is robust to tolerate content-preserving image manipulations, and it has satisfactory generalization capability in different data sets.

Acknowledgments

We would like to thank Dr. Zhaohong Li for her kindness by providing us with their codes. We also would like to acknowledge the helpful comments and kindly suggestions provided by anonymous referees.

References


Table 5

<table>
<thead>
<tr>
<th>Training samples</th>
<th>Testing samples</th>
<th>$P_{TN} (%)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Columbia Image Database</td>
<td>UCID (photographs)</td>
<td>95.00</td>
</tr>
<tr>
<td>Columbia Image Database</td>
<td>UCID, Columbia Image Database</td>
<td>94.00</td>
</tr>
<tr>
<td>UCID, Columbia Image Database</td>
<td>Columbia Image Database (photographs, PRCG images)</td>
<td>95.50</td>
</tr>
<tr>
<td>UCID, Columbia Image Database</td>
<td>Columbia Image Database (photographs)</td>
<td>91.00</td>
</tr>
<tr>
<td>UCID, Columbia Image Database</td>
<td>Columbia Image Database (PRCG images)</td>
<td>94.00</td>
</tr>
<tr>
<td>Art-CG gallery database, Columbia Image Database</td>
<td>Columbia Image Database (photographs, PRCG images)</td>
<td>95.50</td>
</tr>
<tr>
<td>Art-CG gallery database, UCID</td>
<td>Columbia Image Database (photographs, PRCG images)</td>
<td>94.50</td>
</tr>
<tr>
<td>Art-CG gallery database, UCID</td>
<td>Columbia Image Database (PRCG images)</td>
<td>92.50</td>
</tr>
<tr>
<td>Art-CG gallery database, UCID</td>
<td>Columbia Image Database (PRCG images)</td>
<td>93.00</td>
</tr>
</tbody>
</table>


