Different-quality Re-demosaicing in Digital Image Forensics

[1]Bo Wang, [2]Xiangwei Kong, [3]Lanying Wu
[*1,2,3] *School of Information and Communication Engineering, Dalian University of Technology*
E-mail: bowang@dlut.edu.cn, kongxw@dlut.edu.cn, wlydlut@hotmail.com

Abstract

As an important branch of digital image forensics, image splicing is the most fundamental step in photomontage. In the present paper, an efficient blind digital forensics method for image splicing localization and forgery detection is proposed. The method is based on the estimated natural counterpart of the spliced image using different quality re-demosaicing approaches. By comparing the test image with its estimated natural counterpart, the abrupt edges along the spliced region are exposed and a binary image is obtained to illustrate the localization of the splicing. The features extracted from the binary image are fed into a support vector machine classifier to detect spliced forgeries. The DVMM uncompressed spliced image database is used to evaluate the performance of the proposed method. The experimental results show the effectiveness of the method on splicing localization and its accuracy on forgery detection.

Keywords: *Digital Image Forensics, Image Splicing Localization, Re-Demosaicing*

1. Introduction

Photomontage has a history as long as photography, and has become a new serious problem in the digital epoch. During the analog image period, creating an image forgery has been difficult because of the required sophisticated techniques of dark room manipulations. However, image forgeries have become relatively easy in recent years due to the rapid development of digital cameras and high-performance photo-editing software. As a result, forgeries transmitted via the internet have increased and negatively affected many aspects of the society, such as the perception of public trust. A typical example is the widely spread picture of John Kerry and Jane Fonda at an anti-war rally before the presidential year 2004 in the United States. The picture, which has been meant to malign the political life of John Kerry, turned out to be a spliced forgery according to a later report.

More image forgeries currently appear on the internet and public media are challenging the trust of people. Hence, finding solutions to detect the authentication of digital images has become urgent. Digital image forensics is being given increasing attention because it provides a blind and passive approach without embedding advanced information in images.

Image splicing is usually the initial step in creating a forgery. Splicing is done by cropping and pasting image fragments on the same or different images to cover some special objects or adding some additional objects. Splicing is a simple copy-and-paste manipulation without any post-processing such as retouching, contrast adjustment, and lossy compression. However, the sophisticated photo-editing software Adobe Photoshop usually guarantees that the spliced forgeries be unperceivable to the human visual system (HVS). Indeed image splicing detection is a challenging task [1].

To automatically expose a potential spliced image forgery, many passive and blind splicing detection algorithms have been proposed over the past few years. Farid [2] has proposed an approach to expose image manipulations including image splicing based on a statistical model of a “natural” image. Ng and Chang [3] have proposed an image splicing model based on the idea of bipolar signal perturbation, and the bicoherence features are used to detect spliced forgery [4]. Chen *et al*. [5] have introduced two-dimensional (2-D) phase congruency and statistical moments of characteristic functions to digital forensics. They further proposed a natural image model consisting of statistical features, including moments of characteristic functions of wavelet subbands and Markov transition probabilities of difference 2-D arrays [6,7]. With this model, relatively good detection accuracy is obtained compared to the method of Ng and Chang [3, 4]. A further study based on these methods is reported by Christlein *et al*. [8] and Farid [9]. Hsu and Chang [10] have used geometry invariants and camera characteristic consistency to detect spliced images. Moreover, several physical characteristics [11-19]

introduced by the components of the image pipeline, such as the camera response function (CRF), have been used for splicing detection.

Another practical yet important problem is splicing localization. People are concerned more about identifying the location of a spliced region, rather than only knowing whether an image is spliced. Despite this fact few relevant works are reported because splicing localization is challenging. For detecting the duplicated regions in a forged image, Fridrich *et al*. [20] as well as Popescu and Farid [21] propose methods that compute the correlations among fixed-size image blocks. Pan and Lyu [22] have used the scale-invariant feature transform (SIFT) feature to detect image region duplication. They have even localized the scaled and rotated spliced region because of the robustness of SIFT algorithm against scaling and rotation. Amerini *et al*. [23,24] as well as Bravo- Solorio and Nandi [25,26] have proposed similar algorithms. These methods are available for cope-and-paste forgeries, but do not usually work when the spliced region is pasted from another image. Dirik and Memon proposed a block-wised image tampering localization method based on demosaicing artifacts [27], by estimating different levels of MSE (Mean Square Error) between test images and its re-interpolation versions. Qu *et al*. [28] have formulated the shifted double JPEG compression as a noisy convolutive mixing model, and have exposed the tampering region by determining if the image blocks have been compressed twice. Methods for detecting spliced image using image lighting inconsistency have also been proposed [29-31]. The concept of a "digital bullet scratch" for localizing the tampered region in an image has also been suggested [32-34]. Hsu and Chang [35] have proposed a method based on camera function consistency combined with an automatic segmentation algorithm to detect image splicing. Their results indicate that an elementary and incomplete localization of a spliced region can be achieved. However, a more precise splicing localization algorithm is needed because the detection accuracy of the method [35] is usually affected by the empirical segmentation number and texture of images, among other.

In the present paper, we propose an approach for image splicing localization and forgery detection. Different-quality re-demosaicing methods are used to obtain natural counterpart estimations of a test image with different image qualities. The estimated image shows similar statistical characteristics with the natural image, and distinct differences with the spliced image, because of the differences between the consistency of natural image and abrupt transition of spliced edge. Via comparing the smoothness between the test image and its estimated counterpart, the algorithm can provide a credible localization of the spliced region. For distinguishing the forgeries from the authentic photograph, two features are extracted from the splicing localization result, which is a binary image that indicates the region of the forgery. Finally, the features are fed into a classifier. The experimental results are evaluated using the publicly available database DVMM [36], and show that the method can localize the spliced region as well as detect forgeries with a high accuracy.

The rest of this paper is organized as follows. A simple and quick review of demosaicing algorithms is introduced in Section 2. In Section 3, the method for estimating the natural counterpart of the test image with different qualities is proposed. A description of the localization algorithm of the spliced region follows. Post-processing and feature extraction, as well as a simple introduction of the classifier used in our method are also presented in this section. In Section 4, the experimental details are provided. The relationship of the different-quality re-demosaicing algorithms with the performance of the splicing localization are also given therein. Finally, the paper is summarized in Section 5.

2. Demosaicing in digital camera

Theoretically, three CCD/CMOS sensors are required to capture a RGB-colored image in a digital camera to detect red, green, and blue lights, respectively. However, most consumer digital cameras are equipped with just a single CCD/CMOS sensor to capture a mosaicing image for less cost. Therefore, to obtain a RGB image from the mosaicing one, different demosaicing algorithms are used to estimate the missing colors. One of the most important factors affecting image quality is the demosaicing method. In our method, different demosaicing algorithms are used to estimate the different qualities of the natural counterpart of the image. In this section, the image formation pipeline in digital cameras is briefly discussed. Several typical demosaicing algorithms are then introduced.

2.1. Image formation pipeline

Fig. 1 shows the image formation pipeline. When taking a photo of a scene using a digital camera, the photon counts enter the camera through the lens and optical filter. For most consumer-end cameras, there is a color filter array (CFA) placed before the sensor, which is carefully designed according to the HVS. Fig. 2 shows the typical Bayer CFA, which consists of several 2×2 basic units, including one red, one blue, and two green components. By this scheme, only the corresponding color light is allowed to get through the filter array and be captured by the sensor.

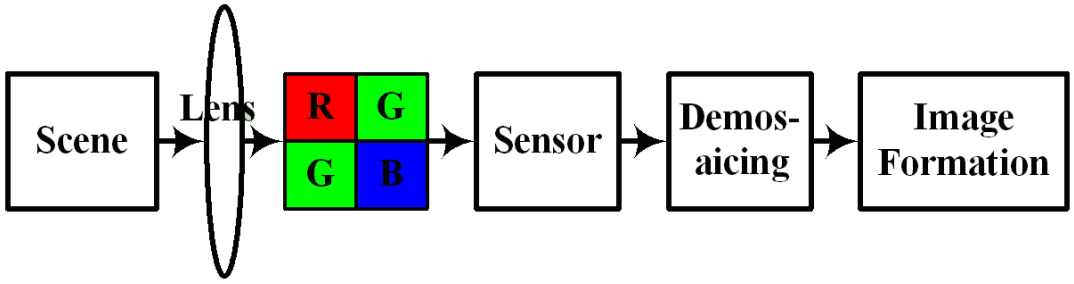

Figure 1. Image formation pipeline

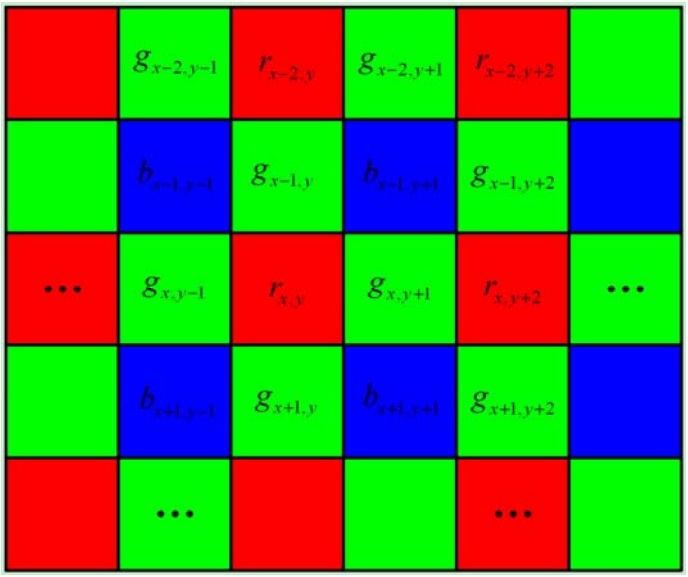

Figure 2. Bayer CFA

The intensity is then converted into voltages, which are subsequently quantized by an A/D converter. The digital signal in each pixel represents the intensity of only one of three colors that is sampled by the CFA. Hence, to obtain a RGB-colored image, an interpolation algorithm is employed to estimate the missing two color components in each pixel using the neighbor-sampled pixel values. After demosaicing, formation processing like white-balancing and gamma correction is conducted. The output image is finally stored in an user-selected image format, which is typically JPEG or TIFF.

2.2. Typical demosaicing algorithms

As described in the last subsection, an important step in image formation pipeline is demosaicing. By demosaicing, a CFA-sampled image is converted to a RGB-colored image. Demosaicing algorithms vary among different camera makers and models. Several typical interpolation algorithms are simply introduced below.

The simplest kernel-based demosaicing algorithms, namely, those that are bilinear and bicubic [37], independently act on each channel. Modeling with a low-pass filter, bilinear algorithm can be represented by the following 2-D filter kernels for the green and red/blue channels:

$$f_{l-g}=\frac{1}{4}\begin{bmatrix}0&1&0\\1&4&1\\0&1&0\end{bmatrix},\quad f_{l-r/b}=\frac{1}{4}\begin{bmatrix}1&2&1\\2&4&2\\1&2&1\end{bmatrix}$$

For the bicubic algorithm, the kernels for the green and red/blur channels are:

$$f_{c-g}=\frac{1}{256}\begin{bmatrix}0&0&0&1&0&0&0\\0&0&-9&0&-9&0&0\\0&-9&0&81&0&-9&0\\1&0&81&256&81&0&1\\0&-9&0&81&0&-9&0\\0&0&-9&0&-9&0&0\\0&0&0&1&0&0&0\end{bmatrix},\quad f_{c-r/b}=\frac{1}{256}\begin{bmatrix}1&0&-9&-16&-9&0&1\\0&0&0&0&0&0&0\\-9&0&81&144&81&0&-9\\-16&0&144&256&144&0&-16\\-9&0&81&144&81&0&-9\\0&0&0&0&0&0&0\\1&0&-9&-16&-9&0&1\end{bmatrix}$$

These non-adaptive bilinear and bicubic interpolation methods exhibit good performances in smooth regions because of their low-pass filter characteristics.

Considering the texture of an image, a gradient-based interpolation algorithm has been proposed by Laroche and Prescott [38]. This method provides an effective way to preserve textures by interpolating along edges. The gradient-based method first computes derivative estimators along the horizontal and vertical directions for the green component. Taking the red sampling pixel $r_{x,y}$ for example, the second-order derivative estimators are generated as follows:

$$H_{x,y}=\left|\left(r_{x,y-2}+r_{x,y+2}\right)/2-r_{x,y}\right| \tag{1}$$

$$V_{x,y}=\left|\left(r_{x-2,y}+r_{x+2,y}\right)/2-r_{x,y}\right| \tag{2}$$

According to the estimators, the missing green component is then interpolated by the nearest green samples in a 3×3 neighborhood along the lower derivative direction. The details of the interpolation are given by the following equation:

$$G_{x,y}=\begin{cases}\left(g_{x-1,y}+g_{x+1,y}\right)/2, & H_{x,y}>V_{x,y}\\ \left(g_{x,y-1}+g_{x,y+1}\right)/2, & H_{x,y}<V_{x,y}\\ \left(g_{x-1,y}+g_{x+1,y}+g_{x,y-1}+g_{x,y+1}\right)/4, & H_{x,y}=V_{x,y}\end{cases} \tag{3}$$

To estimate the missing blue component for pixel $r_{x,y}$, an average bias between the sampled blue pixels and the estimated green pixels is computed in the neighborhood. The bias is then added to the $G_{x,y}$ to generate the blue value, which is shown in Eq. (4):

$$B_{x,y}=\frac{\left(b_{x-1,y-1}-G_{x-1,y-1}\right)+\left(b_{x-1,y+1}-G_{x-1,y+1}\right)+\left(b_{x+1,y-1}-G_{x+1,y-1}\right)+\left(b_{x+1,y+1}-G_{x+1,y+1}\right)}{4}+G_{x,y} \tag{4}$$

A similar manner is applied for the estimation of missing red components. An important feature of gradient-based demosaicing method is the tradeoff between texture preservation and pixel continuity.

Some more complex and better-performance demosaicing algorithms have been recently proposed [39, 40]. Compared with the gradient-based method, these adaptive interpolation algorithms usually employ more complex derivative estimators as classifiers. To ensure that more textures are preserved and accurately describe, there are more directions instead of only horizontal and vertical. One typical kind is the adaptive color plane (ACP) method, which is a modification of the gradient-based method. In the ACP method, the pixel is also first classified using derivative estimators along the horizontal and vertical directions, as follow:

$$H_{x,y} = \left|g_{x,y-1} - g_{x,y+1}\right| + \left|-r_{x,y-2} + 2r_{x,y} - r_{x,y+2}\right| \tag{5}$$

$$V_{x,y} = \left|g_{x-1,y} - g_{x+1,y}\right| + \left|-r_{x-2,y} + 2r_{x,y} - r_{x+2,y}\right| \tag{6}$$

The green value is then estimated using a more complex equation:

$$G_{x,y} = \begin{cases} \left(g_{x-1,y} + g_{x+1,y}\right)/2 + \left(-r_{x-2,y} + 2r_{x,y} - r_{x+2,y}\right)/4, & H_{x,y} > V_{x,y} \\ \left(g_{x,y-1} + g_{x,y+1}\right)/2 + \left(-r_{x,y-2} + 2r_{x,y} - r_{x,y+2}\right)/4, & H_{x,y} < V_{x,y} \\ \left(g_{x-1,y} + g_{x+1,y} + g_{x,y-1} + g_{x,y+1}\right)/4 + \left(-r_{x-2,y} - r_{x,y-2} + 4r_{x,y} - r_{x,y+2} - r_{x+2,y}\right)/8, & H_{x,y} = V_{x,y} \end{cases} \tag{7}$$

The estimation of missing chrominance samples depends on their location in the CFA, which is much more complicated than in the gradient-based algorithm. We would not describe this in detail because of the length limit of the present paper. A more elaborate description can be found in the work of Hamilton and Adams [39]. Another efficient demosaicing algorithm called the threshold-based variable number of gradients method has been proposed by Chang *et al*. [40].

3. Proposed Method

Natural images show smoothness and correlations in small local regions, even at sharp edges of an object. These features are caused by the CFA interpolation algorithm employed in the image formation pipeline. Hence, an output image from a digital camera can be modeled as a smooth signal, and splicing is very likely to introduce discontinuity or abrupt change at the spliced point [3]. Based on this observation, we propose an image splicing detection method by detecting and localizing the discontinuity of pixels in the neighborhood.

The framework of our method is illustrated in Fig. 3. There are four main steps in the proposed method to localize image splicing and detect forgeries. First, the natural counterpart of the image is estimated by re-sampling the image in a Bayer CFA manner and re-demosaicing. The natural counterpart is named as a “natural image”. Afterwards, the test and natural counterpart image are compared to determine the authentic pixels and spliced pixels. According to this comparison, a binary index image with the same size as the test image is generated. With a post-processing, the index image indicates the spliced edges. For the purpose of forgery detection, two features are extracted from the binary image, which is the output of comparison. These features are fed into the classifier as inputs to obtain the training model, which is finally used to classify authentic images and forgeries. In the following subsections, these four steps are described in detail.

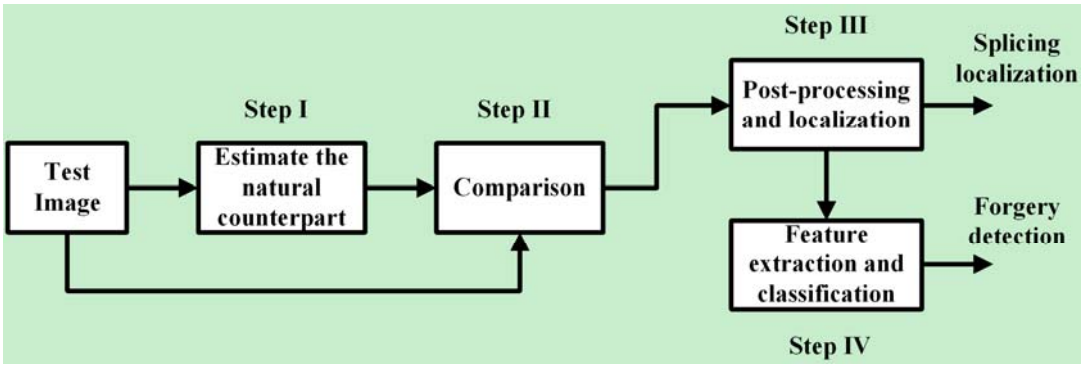

Figure 3. Framework of the proposed method

3.1 Estimated natural counterpart

For a practical scenario, only the suspect image for testing is available. If a natural counterpart of the test image is obtained, then the comparison between the image and its estimated counterpart could reveal the traces along the spliced edge because of the abrupt changes introduced by the splicing. On the other hand, for the authentic image, the estimated counterpart is expected to be approximately similar to the test image. Based on these facts, a good performance estimation of the natural counterpart for the test image is needed.

In the proposed method, CFA re-sampling and re-demosaicing, which are important functions of image formation pipeline in a digital camera, are used to estimate the natural counterpart of the suspect image. As previously described, CFA sampling and demosaicing introduce correlations and continuity to images, whereas splicing very likely destroys the smoothness along the composite edge. By re-demosaicing the suspect image, the continuity of the estimated natural counterpart can be reconstructed. Moreover, the choice of the CFA re-sampling pattern and demosaicing algorithm may greatly influence the estimation result. Hence, the selection must be done carefully.

The key of natural counterpart estimation is to re-build the coherence and consistency in the image neighborhood, therefore we adopt the most popular Bayer CFA, as illustrated in Fig. 2, as the re-sampling pattern in the proposed method. Though there are other 3 patterns of the Bayer CFA arrangement, we consider similar results among these patterns because of their simple but effective sampling ability. The other factor is the demosaicing algorithm, which is diversified and can be divided into two classes according to the image quality obtained. These classes are adaptive and non-adaptive algorithms. Non-adaptive methods include bilinear, bicubic and smooth hue, which has an effect similar with a low-pass filter. The usual result is significant blurring along edge regions. In this case, the comparison may yield a relatively bigger variance along the natural edge regions, and eventually lead to a false positive alarm. Complex adaptive demosaicing methods perfectly preserve edge information. The discontinuity in spliced images is usually maintained, and revealing the estimated counterpart not the “natural” one. As a result, the comparison exposes little differences between the test image and its estimated counterpart, and eventually leads to a false negative alarm. Besides, more demosaicing estimation method are proposed in recent years, a better result is expected while a better demosaicing estimation result is achieved.

Considering the balance between false positive and negative alarms, three different demosaicing algorithms are then adopted as the interpolating method to estimate different-quality “natural” images based on the above analysis. These are bicubic, gradient-based, and ACP algorithm.

Given an $M \times N$ suspect RGB-colored image I_t, I_t can be represented as:

$$I_t = \left\{ p_{x,y,k} \,\middle|\, x \in [1, M], y \in [1, N], k \in \{R, G, B\} \right\} \tag{8}$$

$p_{x,y,k}$ denotes each single pixel in the image. *R*, *G*, and *B* indicate the red, green, and blue components, respectively. The estimation process of the natural counterpart I'_t is shown in Eq. (9) and Fig. 4.

$$I'_t = f_{dem}\left(f_{cfa}\left(I_t\right)\right) \tag{9}$$

where $f_{dem}\left(\cdot\right)$ and $f_{cfa}\left(\cdot\right)$ denote the demosaicing methods and Bayer CFA sampling respectively.

Figure 4. Flow of the natural counterpart estimation in the proposed method

3.2 Comparison

Once the natural counterpart of the test image is obtained, it is compared with the test image to expose the spliced artifacts as previously described. Obvious biases between the spliced image and its counterpart may be clearly seen along the spliced edges, whereas the authentic images are similar with their estimated version.

A comparison is made using the absolute value of the difference between the test image and its counterpart. For each single pixel, three distances are computed for the red, green, and blue components, respectively:

$$d_{x,y,k} = \left| p_{x,y,k} - p'_{x,y,k} \right|_{x\in[1,M],y\in[1,N],k\in\{R,G,B\}} \tag{10}$$

Each distance is compared to a threshold $T_k, k \in \{R,G,B\}$ calculated by Eq. (11) in the corresponding color component. The threshold is the combination of an empirical factor α and the maximum difference between the neighbor pixels along four directions, horizon, vertical, as well as left and right diagonals. By comparison, we assume that if the bias between the suspect pixel and its counterpart exceeds a certain rate across the edge, then splicing may occur. In our method, if any of the three-color components has a distance that exceeds the corresponding threshold for a given pixel, then it is classified as a spliced pixel.

$$T_k = \alpha \times \max\left(\left|p_{x-1,y,k} - p_{x+1,y,k}\right|, \left|p_{x,y-1,k} - p_{x,y+1,k}\right|, \left|p_{x-1,y-1,k} - p_{x+1,y+1,k}\right|, \left|p_{x-1,y+1,k} - p_{x+1,y-1,k}\right|\right) \tag{11}$$

$$p_{x,y} = \begin{cases} 1, & Spliced, \quad if\ \exists d_{x,y,k} > T_k \\ 0, & Authentic, \quad if\ \forall d_{x,y,k} \le T_k \end{cases} \tag{12}$$

The parameter α , which balances the false positive and negative alarms of detection, is experimentally determined as 0.9. Unfortunately, this empirical parameter may differ from different image sets.

As a result of the comparison, a binary image with the same resolution as the test image is obtained. This binary image represents the spliced pixels with 0 and authentic pixels with 1. The raw black-and-

white image I_r, is the primordial result that is used to localize and expose splicing by post-processing and classification.

3.3 Post-processing for splicing localization

The raw result I_r could be used as the primary result for splicing localization. However, there are usually some authentic pixels misclassified as spliced ones. By analysis, the false positive alarm is seen to occur usually in a smooth area because of two reasons. One is the magnified bias between the test pixel and its estimated version. The pixel value needs to be rounded off to the nearest integer to obtain the natural counterpart image after re-demosaicing. Hence, this operation can sometimes magnify the bias compared with the threshold. We call this kind of points as a "flat pixel" because the pixel looks like the bottom of a flat dish. The other reason is noise. Noisy points usually show discontinuity with the neighborhood that would be falsely considered as spliced pixels. To get a more accurate result, the false positive caused by these two abovementioned kinds of pixels needs to be decreased.

For flat pixels, an edge detection algorithm is employed to remove the false positive points in the raw result. In our method, a reasonable assumption is made to ensure that only the pixels located at the edge of the spliced object are the spliced pixels. Those that are not edge points cannot be classified as spliced pixels in our approach. Canny edge detection is applied to detect the edge pixels in the image in our method. This method provides a good approach to detect strong and weak edges, and is therefore less likely to be fooled by noise than the others. Consequently, more edge information can be provided. After the canny edge detection of the test image, an operation of logical AND is applied to the edge detection result and the raw result I_r. The result of this operation is denoted as $I_{r\text{-}e}$. The application of the edge detection method can effectively solve the problem of flat pixels.

To reduce the effect of noisy pixels, a filter for $I_{r\text{-}e}$ is designed. In each 3×3 block of the $I_{r\text{-}e}$, if there are at least two adjacent spliced pixels marked as 1 (Fig. 5a), the spliced pixels in the block are regarded as real forgeries. Otherwise, the spliced pixels are considered as false positive alarms and re-marked as 0 (Fig. 5b and 5c). The result of the filter is denoted as $I_{r\text{-}e\text{-}f}$ after this operation, and most of the false positive alarms caused by the noisy pixels are eliminated.

Figure 5. Samples of (a): real forgery block, as well as (b) and (c) false positive alarms caused by noisy pixels

After the above steps, a binary image $I_{r\text{-}e\text{-}f}$ is obtained. This binary image localizes the spliced edge with white points and authentic pixels with black. Based on the resulting binary image, the splicing location can be exactly determined, and the approximate shape of the forgery object can be detected.

3.4 Feature extraction and classifier for forgery detection

The spliced object can be localized by manually observing the binary image $I_{r\text{-}e\text{-}f}$. However, an auto-classification is sometimes demanded to detect the forgeries. For this automatic exposure, two features are extracted from the binary image, and a classifier is designed in this subsection to distinguish between the spliced and authentic images.

The first feature is simply defined as the ratio of spliced pixel numbers to image size:

$$f_1 = C_{sp} \Big/ \left(M * N \right) \tag{13}$$

where C_{sp} is the count of 1 in the binary image $I_{r\text{-}e\text{-}f}$. Although simple, this feature is the most intuitive presentation of the number of spliced pixels in the test image.

The other feature considers the density of spliced pixels in 3×3 blocks. The key observation that leads to this feature is that for a tampered image, it is expected that the density of spliced pixels in a sub block should be higher than that in authentic image. The binary image is scanned by a non-overlapped 3×3 block, and the number of spliced pixels in each block that indeed contain the spliced pixels is obtained. The mean of the spliced pixels in these blocks, as shown in Eq. (14), is the other feature f_2 that is later fed into the classifier.

$$f_2 = \sum_{i=1}^{C_{sb}} C_{sp-b}^{i} \Big/ C_{sb} \tag{14}$$

where C_{sb} is the count of blocks that contain spliced pixels, and C_{sp-b}^{i} is the count of 1 in the i^{th} spliced 3×3 block.

Considering the classifier, Support Vector Machine (SVM) is applied in our method. SVM is a set of related supervised learning methods used for data classification. A typical SVM classifier works with training and classification steps. In the training process, a set of samples with different classes is given. Several features are extracted from these samples and fed to the SVM as input vectors, with their specified class labels. The input vectors are mapped to a higher dimensional space where a maximal separating hyper plane is constructed. Subsequently, SVM, also known as a maximum margin classifier, converts the problem of classification to a process of solving an optimization problem. The solution simultaneously minimizes the empirical classification error and maximizes the geometric margin. Several certain kernel functions including linear, polynomial, radial basis function (RBF) and sigmoid kernels are applied in such an optimization problem via solving a dual quadratic function. Finally, several parameters are generated as the training model. For the classification step, the features are also extracted from the test samples and fed into the SVM. With the prior trained model, the predicted labels for the test samples are given as the output of the classifier.

Different kernel functions offer different classifiers. We notice that the non-linear kernels allow the algorithms to fit the maximum-margin hyper-plane in the transformed feature space, and thus make linearly non-separable samples as separable as possible. Based on this observation, the *C*-support vector classification (*C*-SVC) [41] is used with one of the basic kernels called the non-linear RBF as the classifier. The parameters C and γ are determined by a grid search using cross validation. In our method, the grid search is applied with C and γ in the same range of $\{2^{-5}, 2^{-4}, \cdots, 2^{5}\}$. This lengthy procedure is not discussed in the present paper.

4. Experiments and discussions

The image dataset used in our experiments for evaluating the performance of the proposed method is provided by DVMM [36]. A total of 183 authentic and 180 spliced images are included in the open authentic/spliced image dataset. The resolutions vary from 757×568 to 1152×768. All the authentic images are taken with 4 cameras (Canon G3, Nikon D70, Canon 350D Reble XT, and Kodak DCS 330). For the spliced image category, they are forged by pasting an authentic image with visually salient objects copied from another image taken by a different camera. Adobe Photoshop is used as the image processing tool. No post-processing was performed on the spliced images. For each camera pair, 30 images are created, resulting in a total of $P_4^2 \times 30 = 180$ images with 4 cameras in the dataset. All the authentic images and spliced forgeries are saved in an uncompressed TIFF format. Fig.6 illustrates the samples of the authentic images and spliced forgeries, and also the counterparts of the images using different-quality soft re-imaging algorithms. There is no obvious difference among different soft re-imaging algorithms, but they still show different characteristics in the image pixel neighborhood.

Fig. 7b, 7c, and 7d illustrate the localization results using different counterpart for comparison. For the authentic images, no evident spliced edge in the binary image is exposed, although false positive alarms occur at some pixels. However, for the spliced forgeries, the localization results show obvious contours of the pasted object with the edges marked in white.

Different-quality soft re-imaging algorithms provide different-quality localization results. The typical non-adaptive interpolation algorithm, the bicubic algorithm, shows similar characteristics with a low-pass filter. Consequently, more edge information is provided in the authentic images and forgeries, and these edges are falsely classified as spliced edge (Fig. 7b). However, the complex adaptive interpolation method, such as the ACP, provides a nearly perfect performance with the authentic image. In contrast, the algorithm for the forgeries misses numerous spliced edge pixels. Hence, the shape of the spliced area is not exact, as Fig. 7d illustrates. In the experiment, the gradient-based interpolation algorithm is found to yield a better balance between false negative and false positive alarms. As a basic adaptive algorithm, the gradient-based method preserves the authentic edges in the test image. Simple interpolations along the horizontal and vertical directions ensure that the possible discontinuity along the spliced area is coherent and natural in the estimated counterpart image. These characteristics guarantee the high accuracy of the classifier on both the authentic image and splicing pixels.

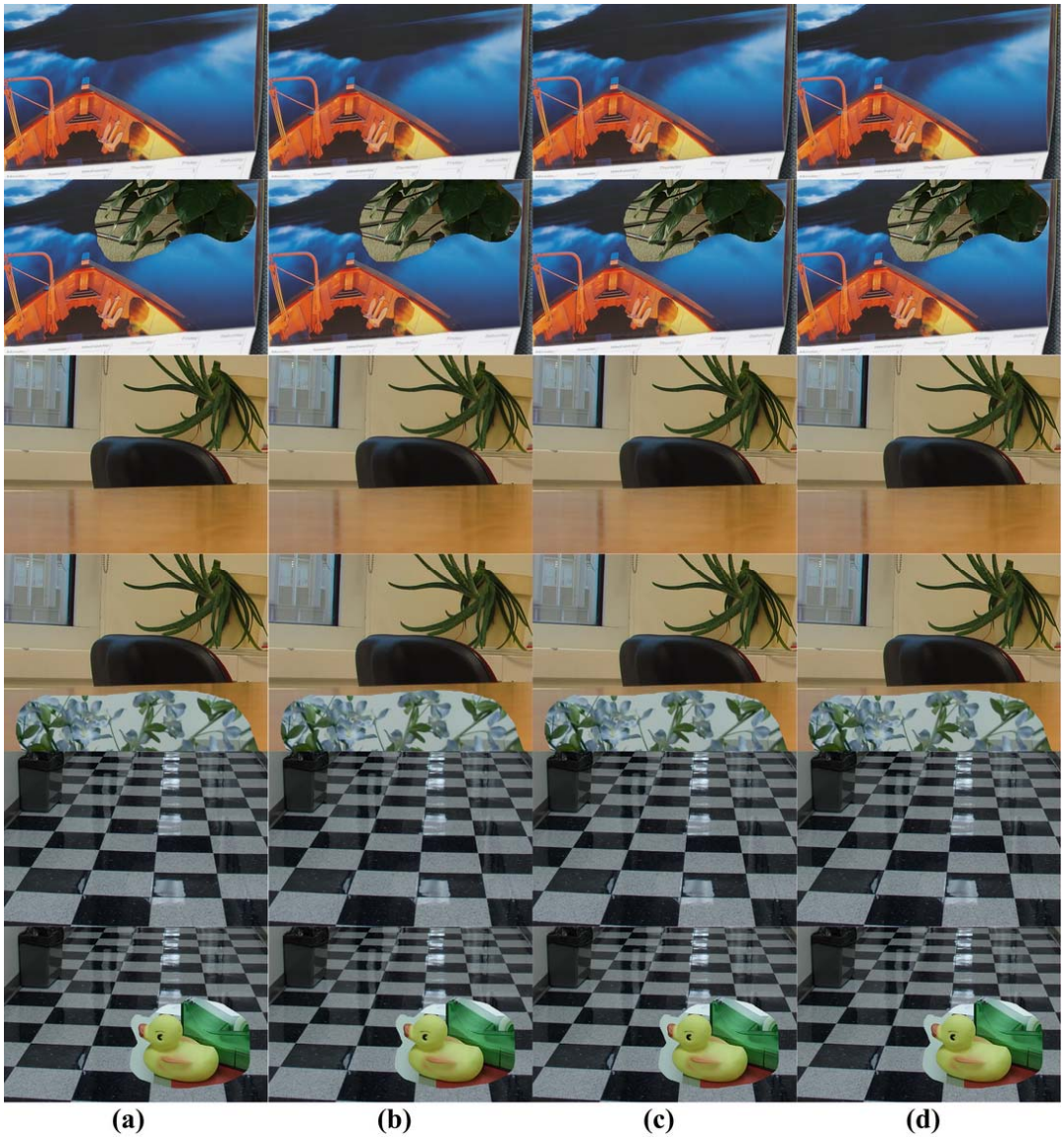

Figure 6. Sample of authentic images and spliced forgeries as well as their counterparts from different-quality soft re-imaging: (a) sample images, (b) bicubic, (c) gradient based, and (d) ACP.

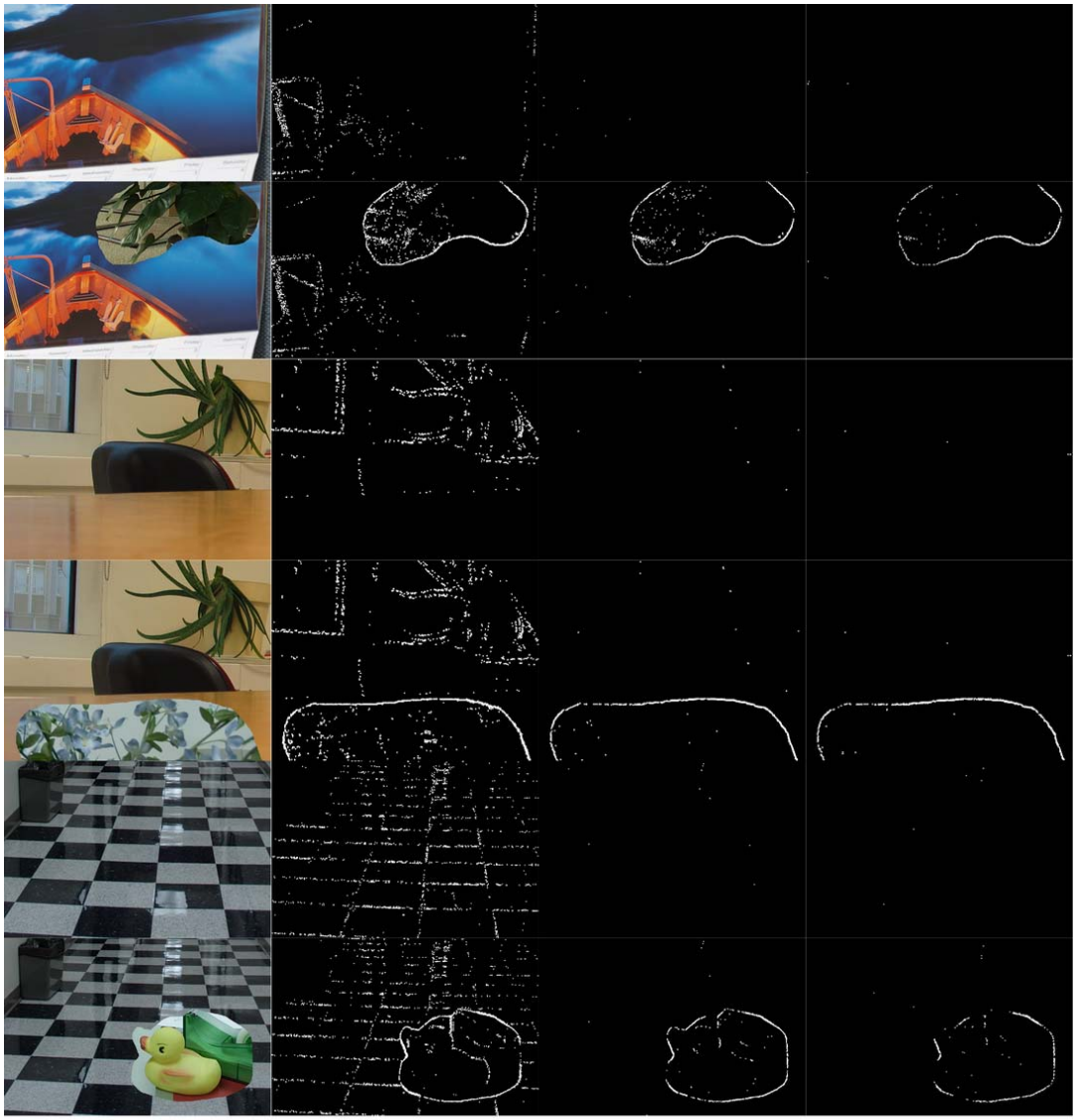

Figure 7. Localization results of authentic images and spliced forgeries using different-quality soft re-imaging: (a) sample images, (b) bicubic, (c) gradient based, and (d) ACP re-imaging.

In the forgery detection experiment, 60 authentic and 60 spliced images are randomly selected as training samples, and the rest of images in DVMM dataset as test samples. The experiment is repeated 20 times to obtain a reliable statistical result. The performances of the methods of Chen *et al*. [5] and Shi *et al*. [7], which are based on RBF SVM, are presented for comparison. In this method, 5/6 of the images are randomly selected for training and 1/6 are for testing to achieve a relatively high accuracy. Table 1 demonstrates the classification results of the presently proposed method and that of Chen's for the same image dataset. In the table, TP denotes the true positive rate, which indicates the accuracy of the classification of authentic images. TN refers to the true negative rate, and corresponds to the detection rate of the spliced images. High detection rates of spliced forgeries for all three different-quality re-demosaicing algorithms are achieved with our method. Regardless of the re-demosaicing method adopted, our method performs better over that of Chen, and similar with that of Shi, in terms of classification accuracy.

Furthermore, the TP rate of our method using the bicubic re-demosaicing algorithm is higher than that using ACP. The inverse is true for the TN rate, as forecasted in the previous section 3.1 of the

present paper. Although the gradient-based re-demosaicing method cannot yield the highest rate for TP and TN, respectively, a better balance between the TP and TN rates is obtained. The highest rate of average accuracy is 89.6%.

Table 1. Statistical detection results and comparison with W. Chen's and Shi's method

Method	W. Chen's	Shi's	Proposed Method		
			bicubic	gradient based	ACP
TP	80.1%	88.4%	82.7%	88.4%	89.3%
TN	81.7%	86.5%	94.4%	90.7%	87.9%
AVG	80.9%	87.5%	88.5%	89.6%	88.6%

5. Conclusions

In the present paper, a novel approach is proposed for localizing spliced areas and exposing image splicing forgeries. For the first time, different-quality soft re-demosaicing methods are introduced for the natural counterpart estimation of a spliced image. The combination of the Bayer CFA and different-quality re-demosaicing algorithms is employed as the estimator. Higher accuracies are achieved using all three different approaches compared with the existing algorithms. However, the gradient-based re-demosaicing method has the best balance between false negative and false positive alarms. This balance is due to its excellent performance in preserving edge information and pixel continuity in the neighborhood. A straightforward comparison is applied to the test image and its estimated counterpart to obtain a binary image. By post-processing, the final result image is obtained, and the spliced area is localized. Two features are extracted from the black-and-white image, and are fed into the SVM classifier to differentiate the authentic images from the spliced ones. The experimental results evaluated via the DVMM image dataset indicate the precision of our method in splicing localization, and the better performance on forgery detection than existing methods.

The key step in our method is the estimator for the counterpart of the test image. With a better estimator, a more precise counterpart of an image can be estimated, and will certainly result in better detection. An extended study on estimator searches shall be conducted by our group in the future.

6. Acknowledgements

This work is supported by the National Natural Science Foundation of China under Grant No. 60971095, and also the Fundamental Research Funds for the Central Universities under Grant No. DUT10RC(3)99.

References

[1] B. L. Shivakumar, S. Santhosh Baboo, "Detecting copy-move forgery in digital images: a survey and analysis of current methods", Global Journal of Computer Science and Technology, vol.10, no.7, pp.61-65, 2011.
[2] Farid H, "A picture tells a thousand lies", New Scientist, vol.179, no.2411, pp.38-41, 2003.
[3] Ng T-T, Chang S-F, "A model for image splicing", IEEE International Conference on Image Processing, pp.1169-1172, 2004.
[4] Ng T-T, Chang S-F, "Blind detection of photomontage using higher order statistics", IEEE International Symposium on Circuits and Systems, pp.688-691, 2004.
[5] Chen W. Shi Y Q, Su W, "Image splicing detection using 2-D phase congruency and statistical moments of characteristic function", SPIE Electronic Imaging, pp.65050R-1, 2007.
[6] Shi Y Q, Chen C, Chen W, "A natural image model approach to splicing detection", ACM Multimedia and Security Workshop, pp.51-62, 2007.
[7] Shi Y Q, Chen C, Xuan G R, Su W, "Steganalysis versus splicing detection", International Workshop on Digital Watermarking, pp.158-172, 2007.

[8] Christlein V, Riess C, Angelopoulou E, “A study on features for the detection of copy-move forgeries”, Information Security Solutions Europe, pp.1-12, 2010.
[9] Farid H. “A survey of image forgery detection”, IEEE Signal Processing Magazine, vol.2, no.26, pp.16-25, 2009.
[10] Hsu Y-F, Chang S-F, “Detecting image splicing using geometry invariants and camera characteristics consistency”, IEEE International Conference Multimedia & Expo, pp.549-552, 2006.
[11] Dong J, Wang W, Tan T, Yun Q. Shi, “Run-length and edge statistics based approach for image splicing detection”, 7th IEEE International Workshop on Digital Watermarking, pp.76-87, 2009.
[12] Zhang J, Zhao Y, Su Y, “A new approach merging markov and DCT features for image splicing detection”, IEEE International Conference on Intelligent Computing and Intelligent Systems, pp.390–394, 2009.
[13] Ng T-T, Chang S-F, Tsui M-P, “Using geometry invariants for camera response function estimation”, IEEE International Conference on Computer Vision and Pattern Recognition, pp.1-8, 2007.
[14] Lin Z, Wang R, Tang X, Shu H-Y, “Detecting doctored images using camera response normality and consistency”, IEEE International Conference on Computer Vision and Pattern Recognition, pp.1087-1092, 2005.
[15] Fu D, Shi Y Q, Su W, “Detection of image splicing based on Hilbert-Huang transform and moments of characteristic functions with wavelet decomposition”, Proceeding of 5th International Workshop on Digital Watermarking, pp.177-187, 2006.
[16] Battiato S, Messina G, “Digital forgery estimation into dct domain: acritical analysis”, Proceedings of the First ACM Workshop on Multimedia in Forensics, pp.37–42, 2009.
[17] Bayram S, TahaSencar H, Memon N, “An efficient and robust method for detecting copy-move forgery”, Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE Computer Society, pp.1053–1056, 2009.
[18] Yanjun C, Tiegang G, Li F, Qunting Y, “A Novel Approach for Detecting Region Duplication in Digital Images”, Advances in Information Sciences and Service Sciences, vol.3, no.10, pp.144-151, 2011.
[19] Michael Z, Sun X, “Digital Image Splicing Detection Based on Local Complexity of Local Maximum Partial Gradient of DWT Coefficients”, International Journal of Digital Content Technology and its Applications, vol.6, no.5, pp.1-9, 2012.
[20] Fridrich J, Soukal D, Lukáš J, “Detection of copy-move forgery in digital images”, Proceeding of Digital Forensic Research Workshop, 2003.
[21] Popescu A, Farid H, “Exposing digital forgeries by detecting duplicated image regions”, Technical Reports TR2004-515, DartmouthCollege, 2004.
[22] Pan X Y, Lyu Siwei, “Detecting image region duplication using SIFT features”, International Conference on Acoustics, Speech, and Signal Processing, pp.1706-1709, 2010.
[23] Amerini I, Ballan L, Caldelli R, Del Bimbo A, Serra G, “Geometric tampering estimation by means of a SIFT-based forensic analysis”, International Conference on Acoustics, Speech, and Signal Processing, pp.1702-1705, 2010.
[24] Amerini I, Ballan L, Caldelli R, Del Bimbo A, Serra G, “A SIFT-based forensic method for copy–move attack detection and transformation recovery”, IEEE Transactions on Information Forensics and Security, vol.6, no.3, pp.1099-1110, 2011.
[25] Bravo-Solorio S, Nandi A K, “Automated detection and localization of duplicated regions affected by reflection, rotation and scaling in image forensics”, Signal Processing, vol.91, no.8, pp.1759-1770, 2011.
[26] Bravo-Solorio S, Nandi A K, “Passive forensic method for detecting duplicated regions affected by reflection, rotation and scaling”, European Signal Processing Conference, pp.824–828, 2009.

[27] Dirik A E, Memon N, “Image tampering detection based on demosaicing artifacts”, IEEE Conference on Image Processing, Cairo, Egypt, pp.1497-1500, 2009.
[28] Qu Z, Luo W, Huang J, “A convolutive mixing model for shifted double JPEG compression with application to passive image authentication”, IEEE International Conference on Acoustics, Speech and Signal Processing,pp.1661-1664, 2008.
[29] Johnson M, Farid H, “Exposing digital forgeries by detecting inconsistencies in lighting”, Proceeding of ACM Multimedia and Security Workshop, pp.1-9, 2005.
[30] Johnson M, Farid H, “Exposing digital forgeries in complex lighting environments”, IEEE Transactions on Information Forensics and Security, vol.2, no.3, pp.450-461, 2007.
[31] Kee E, Farid H, “Exposing digital forgeries form 3-D lighting environments”, IEEE Workshop on Information Forensics and Security, pp.1-6, 2010.
[32] Lukáš J, Fridrich J, Goljan M, “Detecting digital image forgeries using sensor pattern noise”. Proceeding of Security, Steganography, and Watermarking of Multimedia Contents VIII, SPIE Electronic Imaging. pp.362-372, 2006.
[33] Chen M, Fridrich J, Goljan M, Lukáš J, “Determining image origin and integrity using sensor noise”, IEEE Transaction on Information Security and Forensics, vol.3, no.1, pp.74-90, 2008.
[34] Fridrich J, “Digital image forensic using sensor noise”, IEEE Signal Processing Magazine, vol.26, no.2, pp.26-37, 2009.
[35] Hsu Y-F, Chang S-F, “Image splicing detection using camera response function consistency and automatic segmentation”, IEEE international Conference on Multimedia & Expo, pp.28-31, 2007.
[36] Columbia Uncompressed Image Splicing Detection Evaluation Dataset. http://www.ee.columbia.edu/ln/dvmm/downloads/AuthSplicedDataplicedDataSet/uthSplicedDataSet.htm .
[37] Keys R G, “Cubic convolution interpolation for digital image processing”. IEEE Transactions on Acoustics, Speech, and Signal Processing, vol.29, no.6, pp.1153-1160, 1981.
[38] Laroche C A, Prescott M A, “Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients”, US Patent, 5373322, 1994.
[39] Hamilton J F, Adams J E, “Adaptive color plane interpolation in single sensor color electronic camera”, US Patent, 5629734, 1997.
[40] Chang E, Cheung S, Pan D Y, “Color filter array recovery using a threshold-based variable number of gradients, Sensors, Cameras, and Applications for Digital Photography”, Proceedings of the SPIE, pp.36-43, 1999.
[41] Chang C C, Lin C J, LIBSVM: A library for support vector machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm .