IEICE TRANS. INF. & SYST., VOL.E98–D, NO.7 JULY 2015

LETTER

Expose Spliced Photographic Basing on Boundary and Noise Features

Jun HOU[†a)], ***Member and*** **Yan CHENG**[††], ***Nonmember***

SUMMARY The paper proposes an algorithm to expose spliced photographs. Firstly, a graph-based segmentation, which defines a predictor to measure boundary evidence between two neighbor regions, is used to make greedy decision. Then the algorithm gets prediction error image using non-negative linear least-square prediction. For each pair of segmented neighbor regions, the proposed algorithm gathers their statistic features and calculates features of gray level co-occurrence matrix. K-means clustering is applied to create a dictionary, and the vector quantization histogram is taken as the result vector with fixed length. For a tampered image, its noise satisfies Gaussian distribution with zero mean. The proposed method checks the similarity between noise distribution and a zero-mean Gaussian distribution, and follows with the local flatness and texture measurement. Finally, all features are fed to a support vector machine classifier. The algorithm has low computational cost. Experiments show its effectiveness in exposing forgery.

key words: *image splicing, K-means clustering, distribution similarity, flatness measure, texture feature*

1. Introduction

With the advent of low-cost digital camera and editing software, more and more synthetic images become available. This brings a new challenge in many applications, such as criminal investigation, journalistic reporting, and etc. Splicing is one of the most common tampering tricks. If performed carefully, the border between two merged parts is imperceptible. One of the ways to discriminate between original and forged images is based on camera features, such as camera filter array(CFA), camera response function(CRF) and sensor noise. Approach in [1] derives the tampering probability of each 2x2 image block on a new statistical model, without requiring a priori the position of the forged region. An earlier work [2] estimates neighbor pixel value and extracts features of prediction error to disclose forgery. Hsu and Chang [3] estimate CRF from each region using geometric invariants from locally planar irradiance points. CRF-based cross fitting and local image features are computed and fed to support vector machine(SVM). Fridrich [4] estimates the parameters of the noise model from the original camera, and then calculates the correlations between the estimated camera noise and the extracted image noise to authenticate an image. Other effective features to expose forgery include inconsistencies in the illumination [5], light direction, anomalous height ratio of two objects in an image [6], etc.

In this work, we propose a passive detection algorithm to expose splicing forgery. The rest of this paper is organized as follows. Section 2–4 present details of the proposed algorithm. In Sect. 2, proposed boundary features basing on prediction error and image texture are explained. Starting from output image model from a sensor, the proposed noise features are described in Sect. 3. All these features are fed to SVM in Sect. 4. The performance of the algorithm is evaluated in Sect. 5. Finally, we conclude the paper in Sect. 6.

2. Boundary Feature Extraction

2.1 Image Segmentation

Splicing changes the statistical features along boundary. Exploring features along boundaries to find traces of forgery is an effective way. Splicing introduces similar or total different images into the origin one. Therefore, image segmentation can fast locate the spliced part approximately. Segmentation should capture perceptually important grouping or regions, reflect those global aspects and capture perceptually important non-local features. Here the graph-based method [7] is applied. The reader is referred to [7] for more details in relation to image segmentation.

2.2 Statistical Features on Prediction Error

A pixel's value can be predicted according to those of its neighbors. When a photographic is touched, prediction error along boundary also shows different characteristics from that of untouched one, because of camera interpolation operation with the use of CFA. Due to the complexity of real scenes, it's unlikely that edges along spliced parts would align directly with the replaced ones, which would cause abnormality when predicting CFA interpolation. Moreover, spliced edges are usually post-processed, such as smoothing sharp edges or contrarily sharpening, to make spliced parts visual comfort. These operations also change the characteristics of prediction result. In general, prediction error might be caused by both edges and noise. Here an accurate method with limitation condition, non-negative least-square(NGLS) adaptation is used to predict neighbor pixels. NGLS-based

Manuscript received November 17, 2014.
Manuscript revised February 26, 2015.
Manuscript publicized April 1, 2015.
[†]The author is with Shanghai Key Lab of Modern Optical System, University of Shanghai for Science and Technology, 200092, China.
[††]The author is with East China University of Political Science and Law, Shanghai, 200424, China.
a) E-mail: houjun@usst.edu.cn (Corresponding author)
DOI: 10.1587/transinf.2014EDL8232

approach locally optimizes the prediction coefficients inside a causal window to minimize the mean squared error(MSE) under limitation. Compared to least square adaptation used in [2], NGLS has more accurate approximation to the CFA.

Let p_i denote the value of ith pixel in the image I. We predict ith pixel's value in a given causal window, using linear model $\hat{p}_i = \sum_{k=1}^{8} w_k n_{i,k}$ on its eight-connected neighbors $\{n_{i,1}, n_{i,2}, \ldots n_{i,8}\}$, which can also be denoted by vector $\vec{n}_i$. For the kth (k =1,2,...,8) neighbor of ith pixel, its value is denoted by $n_{i,k}$, and w_k is a non-negative weighting factor associated with $n_{i,k}$. Given a casual window size of $W_p \times W_p$ with M pixels(M=$W_p \times W_p$), we present those M pixels as a M×1 column vector $\vec{p} = [p_1, p_2, \ldots, p_M]^T$. All neighbors, used for predicting each pixel in $\vec{p}$, form the M×8 matrix $\vec{n}$.

The weighting coefficients can be achieved by solving a non-negative least-square problem, i.e., $\min_w \left\| \vec{n}\vec{w} - \vec{p} \right\|^2$, subject to that every element in $\vec{w}$ is no less than zero. $\vec{w}$ is the 8×1 weight coefficient vector $\{w_1, w_2, \ldots .w_8\}^T$. Here projected gradient method [8] is applied to find $\vec{w}$. And the prediction value for the ith pixel is denoted as: $\hat{p}_i = \vec{n}_i\vec{w}$. Let $\Delta p_i = |\hat{p}_i - p_i|$ denote the absolute prediction error for ith pixel. For two neighbor regions, we further divide them into three parts, two inner parts and one boundary part between them. Boundary part refers to boundary along segmentation edge, with the width W_b. Two inner parts are two segmented neighbor regions excluding the boundary part. Calculate statistical features of Δp_i in three parts separately. Denote the mean of Δp_i in each part as m_1, m_2 and m_b separately. The variance of Δp_i in each part is represented as v_1, v_2 and v_b respectively. Subscripts b indicates that it is the feature of boundary part. Calculate $\max(m_1, m_2, m_b)$, $\min(m_1, m_2, m_b)$, $\frac{m_1+m_2}{2\sqrt{m_1 m_2}}$, $\frac{m_b}{m_1+m_2}$, $\max(v_1, v_2, v_b)$, $\min(v_1, v_2, v_b)$, $\frac{v_1+v_2}{2\sqrt{v_1 v_2}}$, $\frac{v_b}{v_1+v_2}$.

Get the prediction error image I_p, in which ith pixel has value Δp_i. Then calculate its corresponding histogram in three parts respectively. The normalized histogram is a representation of density for prediction error. $h(\Delta p_i)$ denotes the density of prediction error Δp_i. For two neighbour regions, calculate the mean and variance of histogram in each part, and denote them as m_{h1}, m_{h2}, m_{hb}, v_{h1}, v_{h2}, and v_{hb} respectively. The entropies of Δp_i, denoted by E_{n1}, E_{n2}, and E_{nb} in three part respectively, are also extracted features. Here is:

$$E_n = -\sum_{\Delta p_i} h(\Delta p_i) \log h(\Delta p_i) \tag{1}$$

Prediction error often occurs along boundary. To quantify the energy distribution of prediction error image, define the similarity measurement between prediction error and edge image as following:

$$S_{im} = \frac{\left\| I_{pe}^j \times I_{bm}^j \right\|}{N_b^j \sqrt{\left\| I_{pe}^j \right\|_2}} \tag{2}$$

I_{pe}^j is a part of I_p. It's the result of I_p masked by jth pair of neighbour regions. I_{bm}^j is the binary boundary map for jth pair of neighbour regions, in which a pixel equals to 1 if it belongs to the boundary part and has value 0 otherwise . N_b^j is the number of pixels that is 1 in I_{bm}^j.

2.3 Grey Level Co-occurrence Matrix(GLCM)

Splicing, disordering CFA interpolation rule, affects preceding features. However, image texture characteristic also have impact on preceding features. Here we take texture features into consideration when checking prediction abnormality. In each pair of neighbor regions, GLCMs for three parts are calculated. Form GLCM using offsets sweeping through 0, 45, 90, and 135 degrees respectively. To deduce the size of GLCM, quantize levels to 16. For each part, get 4 GLCMs size of 16×16. Check contrast and entropy of each GLCM. i and j are coordinate of GLCM, c_{ij} is the value in GLCM with coordinate (i, j) . The contrast is defined as:

$$\sum_{n=0}^{15} n^2 [\sum_i \sum_j (i-j)^2 c_{ij}], \text{where } |i-j| = n \tag{3}$$

The entropy of a GLCM is defined as:

$$-\sum_i \sum_j c_{ij} \log c_{ij} \tag{4}$$

2.4 Vocabulary Construction

For every pair of segmented neighbor regions, there are 42 features extracted. The total number of features for a prediction error image varies depending on segmentation result. To get features with constant length, here feature dictionary is used. Subdivide the training data into feature vectors from untouched and spliced pairs. Each group is clustered into n clusters by k-means algorithm [9]. Thus there are $2n$ words in a feature dictionary. For each pair of segmented regions, their feature vector is represented by a nearest word in the dictionary, with Euclidean distance as a measurement. A histogram of word counts with $2n$ bins represents the distribution of feature vectors in an image. Normalize the histogram and take it as a vector with $2n$ elements. By this way, a vector with fixed length is achieved.

3. Statistical Feature Extraction in Noise

Both splicing operation and smoothing/sharpening operation along edges would cause preceding features abnormal. We should further figure out whether splicing operation is a major cause. Fridrich [4] models the output image I from a sensor as:

$$I = g^\gamma .[(1+K)Y + \Lambda]^\gamma + \theta_q \tag{5}$$

where g is the color channel gain, which adjusts the pixel's intensity level according to the sensitivity of the pixel to obtain white balance. γ is the gamma correction factor. And K is a unique stochastic fingerprint of imaging sensors. Y denotes the incident light intensity. Λ is combination of noise, such as dark current, shot noise and readout noise. θ_q represents quantization noise. For Eq. (5), basing on Taylor expansion and keeping the first two terms, there is:

$$I = (gY)^{\gamma} + (gY)^{\gamma}K + \theta \tag{6}$$

θ is a complex of independent random noise components. $(gY)^{\gamma}$ denotes the noise-free sensor output. Suppress the noise-free image by subtracting it from both sides and get:

$$I_f = I - (gY)^{\gamma'} = I' + n_f \tag{7}$$

$(gY)^{\gamma'}$ is the estimation of $(gY)^{\gamma}$. I' is something about K, connected with camera-response. n_f, the combination noise term, is gaussian noise with zero mean. If an image is altered, I' from other camera is a weak signal and n_f dominates the contribution of I_f. Therefore, we consider I' included in the noise term when conducting research on I_f distribution. Oppositely, if an image is unaltered, the distribution of I_f depends on I' and n_f as Eq. (7) shows. The research in [4] gives us a revelation: I_f in an altered images fits Gaussian distribution with zero mean better, compared to that in an authentic image. Here $(gY)^{\gamma'}$ is considered as denoised I using scale mixtures of Gaussians in the wavelet domain [10]. B_{bm} is the mask of boundary image, i.e., B_{bm} is OR result of all I^j_{bm}. Let B_b denote the set of all pixels belonging to boundary parts. Analyse the statistical distribution corresponding to pixel $p_i (p_i \in B_b)$ in I_f. Then check the distribution with a zero-mean Gaussian one, and exam the similarity between them. Let $\hat{p}_f(k)$ be the probability mass function of I_f fit, modeled as follows for simplicity: $\hat{p}_f(k) = d\exp(\lambda|k|^{\nu})$, where d is a normalization constant. $\lambda < 0$, and ν is an exponent parameter. Undoubtedly, linear least-squares(LLS) can be used to solve for d,ν and λ. But LLS is too sensitive for small perturbation. Here we use nonlinear least-squares method to obtain the best fit distribution. Hessian modification is applied here to calculate the newton direction. Also incorporate equality constraints on d, ν and λ using log-barrier function:

$$\begin{aligned}&\min_{d,\lambda,\nu} \sum_k (p_f(k) - d\exp(\lambda|k|^{\nu}))^2 \\ &-\mu(\log(d) + \log(-\lambda) + \log(\nu))\end{aligned} \tag{8}$$

where μ is a regulation parameter. $p_f(k)$ denotes actual I_f distribution in boundary parts. Define:

$$C_{or} = \frac{\left\|\hat{p}_f(k) * p_f(k)\right\|}{\sqrt{\left\|\hat{p}_f(k)\right\| * \left\|p_f(k)\right\|}} \tag{9}$$

C_{or}, the similarity measure between $p_f(k)$ and its Gaussian estimation $\hat{p}_f(k)$, is the first extracted feature in this section.

The variance of n_f, denoted as σ_n^2 varies depending on image texture. For relatively flat region, each region could be considered having same σ_n^2, i.e., n_f satisfies white Gaussian noise, and σ_n^2 usually has low value. For highly textured regions, σ_n^2 is different, i.e., n_f is colored Gaussian noise, and σ_n^2 is higher [4] . Whether n_f is colored has influence on similarity result above. Restore high-frequency image from wavelet transform of HL, LH and HH subband in I_f, and then mask it with binary boundary map B_{bm} to get high-frequency boundary image H_f. Define

$$f_T = \frac{1}{S_b} \sum_{p_i \in B_b} V_{ar\_z}(H_f(i)) \tag{10}$$

where S_b is the total number of pixels whose value in B_{bm} is 1. f_T measures local texture in boundary parts. $V_{ar\_z}(H_f(i))$ is the variance of H_f in $z \times z$ neighbor of the pixel p_i. Here set z to 3, 5 and 7 respectively. f_T is the second feature extracted in this section.

Restore low-frequency image from wavelet transform LL subband of I_f and then mask with B_{bm} to get L_f. We also define an assessment of local flatness:

$$f_F = \frac{1}{S_b} \sum_{p_i \in B_b} V_{ar\_z}(L_f(i)) \tag{11}$$

$V_{ar\_z}(L_f(i))$ is the variance of L_f in $z \times z$ neighbor of the pixel p_i. Set z to 3, 5 and 7 respectively. f_F is the third feature extracted in this section.

Fridrich's algorithm in the work [4], which divides an image into blocks and estimates the noise parameter in blocks to verify their consistency, has a requirement that spliced parts are from cameras of different brands. If spliced parts are from cameras of same brand, the method fails. The proposed method in this paper is free from this precondition because it only checks noise features along boundary parts. As noted in Sect. 2, the CFA alignment is unlikely for spliced parts.

4. Classifier

For each image, the proposed algorithm gets features with fixed length. Feed all features above to a SVM with radial basis function(RBF) kernel to classify authentic images and tampered ones. We use grid searching to select parameters for the classifier.

5. Simulations

We construct image dataset to verify the proposed algorithm. We use camera dataset with images from Canon, Olympus, Nikon and kodak. These images are in JPEG format. Besides, other images are from iphone and Samsung mobile phone. All images are captured completely random with different scenarios and lighting conditions. All tampered images are touched by photoshop. Copy region from one image and paste to another to construct a spliced image. Spliced region can be arbitrary. The forgery set is further divided into two subsets: (I)forgery images whose spliced parts have no further operation following splicing; (II)images whose spliced parts undergo post-processing. The training samples are randomly selected from the image dataset. We randomly choose 800 authentic images and 1200 tampered ones for training, and test on all remaining images. 600 forgery images in subset I and 600 ones from subset II, consist of forgery training images. Set the parameter W_b in Sect. 2.2 to 17. In Sect. 2.4, n equals to 15. The fraction of correctly classified tampered images Pd, and the percentage of authentic images wrongly classified

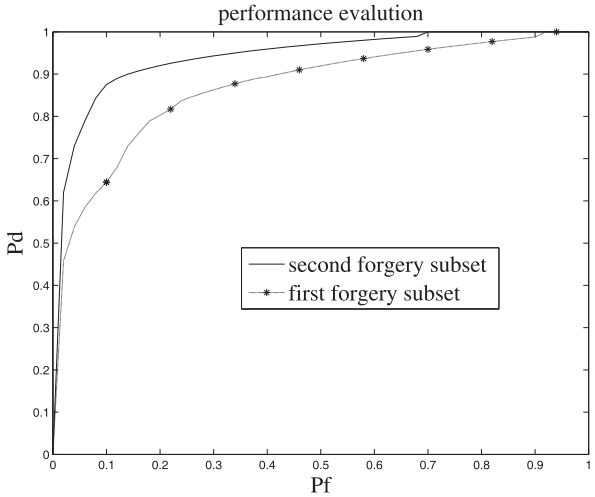

Fig. 1 ROC of two forgery subsets with proposed features

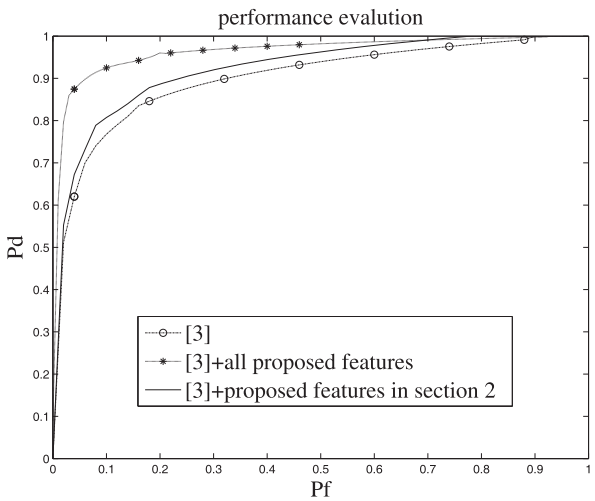

Fig. 2 ROC of first forgery subsets

as tampered Pf are computed to obtain the receiver operating characteristics (ROC). We study ROC on two forgery subsets separately. Figure 1 shows ROC of the proposed algorithm. Our forensic detector provides accurate detections for those spliced images. The method is more efficient in detecting those spliced images whose edges are post-processed to conceal visual perceptibility.

To explore the efficiency of extracted features, we add proposed features to features in [3], and then verify the combined features with forgery subset I. Result in Fig.2 indicates that proposed features increase detection performance. For splicing subset I, features in [3] together with features in Sect. 2 can increase *Pd* slightly. Features in [3] and all proposed features can improve ROC dramatically. Figure 2 also indicates the effectiveness of noise features in Sect. 3.

6. Conclusions

The paper addresses how to expose photographic splicing forgery. Firstly, we segment a photo to capture only those perceptually important aspects of complex imagery. Then pixels in the image are predicted using NGLS method, according to their neighbors. Prediction error often occurs along boundary or rich texture region. With consideration of CFA interpolation, an unaltered image indicates different prediction error statistical features from a spliced one. The difference becomes more obviously if the forgery one is post-processed along boundary. K-means clustering is used to construct dictionary for vector quantization(VQ), and VQ is applied for all features above in each pair of two neighbor region. The histogram of word counts is a vector with fixed length, which would be fed to SVM together with other features. As a contrast to prior work [2], in the proposed algorithm SVM classifier runs only once and therefore the method has lower computation labor. For a spliced image, noise distribution is a Gaussian one with zero mean. The proposed algorithm fits noise image with Gaussian distribution and then measures their similarity, together with flatness and texture measurement. All features above are finally fed to a SVM classifier with RBF kernel. Experiments indicate that the proposed method is effective in disclosing spliced photographic. Proposed features can also be used together with other features in previous literatures, to further improve the fraction of correctly classified tampered images.

References

[1] P. Ferrara, T. Bianchi, A.D. Rosa, and A. Piva, "Image forgery localization via fine-grained analysis of CFA artifacts," IEEE Trans. Inf. Forensics Security, vol.7, no.5, pp.1566–1577, 2012.

[2] J. Hou, H. Shi, Y. Cheng, and R. Li, "Fogery image splicing detection by abnormal prediction features," Proc. IEEE Conf. on Machatronics and Automation, vol.2, pp.1394–1397, 2013.

[3] Y.F. Hsu and S.F. Chang, "Camera response functions for image forensics: an automatic algorithm for splicing detection," IEEE Trans. Inf. Forensics Security, vol.5, no.4, pp.816–825, Dec. 2010.

[4] J. Fridrich, Digital image forensics: there is more to a picture than meets the eye, Springer, 2012.

[5] T.J. Carvalho, C. Riess, E. Angelopoulou, H. Pedrini, and A.R. Rocha, "Exposing digital image forgeries by illumination color classification," IEEE Trans. Information Forensics and Security, vol.8, no.7, pp.1182–1194, 2013.

[6] H. Yao, S. Wang, Y. Zhao, and X. Zhang, "Detecting image forgery using perspective constraints," IEEE Signal Process. Lett., vol.19, no.3, pp.123–126, 2012.

[7] P.F. Felzenszwalb, "Efficient graph-based image segmentation," International J. Computer Vision, vol.59, no.2, pp.167–181, 2004.

[8] C.-J. Lin, "Projected gradient methods for nonnegative matrix factorization," Neural Computation, vol.19, no.10, pp.2756–2779, 2007.

[9] R. Fisher, T. Breckon, K.D. Howe, A.W. Fitzgibbon, C. Robertson, E. Trucco, and C.K.I. Williams, Dictionary of computer vision and image processing, Second ed., John Wiley & Sons, 2013

[10] J. Portilla, A. Strela, M.J. Wainwright, and E.P. Simoncelli, "Image denoising using scale mixtures of gaussians in the wavelet domain," IEEE Trans. Image Process., vol.12, no.11, pp.1338–1351, Nov. 2003