Splicing detection based on improved FISH descriptors

Andrey Kuznetsov
Samara National Research University
Samara, Russia
kuznetsoff.andrey@gmail.com

Artem Lanin
Samara National Research University
Samara, Russia
tesstorm@yandex.ru

***Abstract*—Fake images are becoming more common in the modern world due to the availability of high quality and easy to use tools for falsification. Influencing our opinion about a person and spreading false information they may cause considerable damage. To detect such images and counteract their spreading reliable automatic detectors are needed. This paper presents a method for detecting face splicing using computer vision, based on a comparison of the illumination parameters of faces in a single image. We developed an automatic face splicing detector based on this method and tested its performance on synthesized faces, real faces with controlled lighting, pristine and spliced real images and images processed by equalization. Results of experiments showed that it can be used to help in determining the authenticity of an image, but the presence of several light sources, surfaces with high reflectivity and image post processing performed by criminal may reduce its quality.**

Keywords—image, splicing, detection, forensics, lighting, authenticity.

I. INTRODUCTION

Due to advances in photo editing software even low-skilled users can easily create a fake image that is extremely difficult to distinguish from an original without any instruments. In current reality such fake can have a strong impact on society and lead to critical consequences for persons or companies. In particular, face splicing, an artificial distortion of an image in which a person cut from another image is inserted into an original image, can cause high damage to person's reputation and spread fake information. An example of splicing is shown in Fig. 1. Therefore, the issue of creating algorithms that are able to determine the authenticity of images and counteract the spread of fakes is relevant [1].

To date, several methods for detecting artificial distortions have been proposed. A digital watermark or a digital signature can be embedded in the original image, but despite their high efficiency, they must be created either by a camera or by a person who processes the image. That is impractical for most cases. In other methods preliminary information about the original image is not required. They use the fact that falsification operations lead to statistical changes in digital images or leave some traces that can be used to detect fake. Existing methods of detection find noise inconsistencies that arise after interference in some region of the image [2], specific traces of image transformations [3] and traces of camera components remaining in the output image [4]. In addition, physical level of the scene represented in the image can be analyzed to detect artificial distortions. Methods based on it find inconsistencies in geometry [5], shadows [6] and lighting [7, 8]. They are more resistant to image transformations since signal level traces can be spoiled by such operations as resizing and compression.

Researchers from the University of Florence [9] presented one of such methods based on the use of FISH (Face Intensity-Shape Histogram) descriptors and designed to find face splicing. In this method to determine the presence of artificial distortions a degree of lighting inconsistency of two different faces in one image is estimated. For this purpose, histograms that represent the interaction of faces with light are built for each face in the image. In this paper we developed an algorithm for automatic face splicing detection based on using FISH descriptors.

Fig. 1. Example of face splicing.

II. AUTOMATIC SPLICING DETECTOR STRUCTURE

The developed algorithm takes as input an image with at least 2 faces in it and in the output gives a value by which we can determine whether the image was spliced or not. The algorithm can be divided into 3 parts:

- Face detection.
- Building a 3D model of face and calculating normal vectors.
- FISH descriptors extraction and comparison.

Also, we can consider a number of features that distinguish it from other similar algorithms:

- Using of histograms that have proven effective in many computer vision tasks.
- Dependence only on local image statistics, without a specific mathematical model, which makes the algorithm more effective with real images.
- Faster calculations because of using of histograms and the ability to control the size of feature vectors that are calculated using histograms.

However, this detector has some limitations in use:

- At least two faces must be presented in the image.

- The algorithm does not determine which of two faces is spliced.
- The algorithm does not work well if the scene in the image deviates significantly from the assumptions.
- Image resolution should be high enough to build a 3D model.

The splicing detector structure is shown in Fig. 2. To implement face detector we used the dlib library from Python. At first stage, image areas in which faces were found are extracted. Next, coordinates of 68 landmarks corresponding to specific parts of the face are found for each face. They are used in the next step.

To build a 3D model of face and calculate face surface normal vectors 3DMM (3D Morphable Model) was used. Using 68 landmarks and the image of face, it's 3D model is built and normal vectors are found on it. Each normal vector is associated with an image pixel. Matrices of pixels and normal vectors are used to calculate descriptors in the next step.

III. Calculating Descriptors

When using FISH descriptors we assume that the surfaces are convex, Lambertian (further improvement of the developed version is associated with the use of other mathematical models of illumination, namely the Phong model and the Blinn-Phong model) and sources of light are far from the scene. Thus, intensity values of image pixels will depend only on face surface normals. Therefore, areas that are not suitable for these assumptions (neck, ears, lips, eyes and eyebrows, as well as too light and too dark areas of the skin) are removed from the image. In the original work [9], authors use the brightness range to cut off the necessary areas of the face by the threshold. In this paper, a convolutional neural network is used, which allows to detect elements of face.

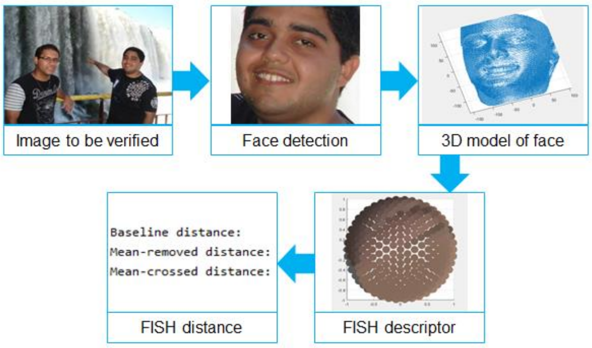

Fig. 2. Automatic splicing detector structure.

Next, a hemisphere containing 305 cells is constructed. This is the FISH descriptor. Each cell corresponds to a certain direction and its value depends on an intensity of pixels in which normal vectors codirectional with the cell are located:

$$I_i = \sum_k \frac{\omega_{ik}}{\omega_i} \hat{I}_k \quad (1)$$

For this we calculate cell weights which are the sum of weights of normal vectors:

$$\omega_i = \sum_k \omega_{ik} \quad (2)$$

Weight of each normal vector relative to each cell is computed from the Gaussian distribution where the standard deviation is equal to 3/8 times the average angular distance between two adjacent cells:

$$z_{ik} = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-1}{2}\left(\frac{arccos(n_i \times \hat{n}_k)}{2\sigma}\right)^2} \quad (3)$$

Vectors with weights that do not exceed the threshold corresponding to 2.5% of the distribution of all weights of the cell are discarded.

$$\omega_{ik} = \begin{cases} z_{ik}, if\, z_{ik} > \tau_k \\ 0, otherwise \end{cases} \quad (4)$$

Finally, combining all color channels we get a FISH descriptor:

$$I_i = [I_R(n_i) I_G(n_i) I_B(n_i)]^T \quad (5)$$

These descriptors are used to compare two faces in the image and get FISH distance. Comparison occurs using the Euclidean norm according to the following equation:

$$D(a,b) = \left(\sum_{i=0,\dots,304} \left\| I_i^a - I_i^b \right\|^2\right)^{\frac{1}{2}} \quad (6)$$

In order to exclude the influence of skin color on the value of FISH distance we can normalize descriptors by average RGB value of face image before calculating (6):

$$\hat{I}_i = I_i / \mu \quad (7)$$

We can also use the second normalization method which takes into account average RGB values of both faces:

$$I_i^{a\to b} = min\left(255, L^a(n_i)\frac{\mu_b}{\mu_a}\right) \quad (8)$$

After that, FISH distance can be calculated as follows:

$$D(a,b) = min\left(D\left(I_i^a, I_i^{b\to a}\right), D\left(I_i^b, I_i^{a\to b}\right)\right) \quad (9)$$

In accordance with a certain threshold and the value of the obtained FISH distance a conclusion is drawn about the presence of artificial distortions.

IV. Experiments

In order to evaluate the effectiveness of the splicing detector, it was tested on various images. To do this we used datasets of synthesized faces Syn1 and Syn2 [10], a dataset of real faces with controlled lighting ExtendedYaleB [11] and a dataset of pristine and spliced real images DSO-1 [12] (Fig. 3). In addition, tests were performed on real images processed by the CLAHE algorithm in order to evaluate the impact of post-processing on detector performance.

Results of these tests in the form of ROC curves are presented in Fig. 4, where red graph corresponds to the FISH descriptor without normalization, green graph – to the FISH descriptor from (7) and blue graph – to the FISH descriptor from (8).

Datasets Syn1, Syn2 and ExtendedYaleB represent images of faces with different lighting conditions. In order to simulate cases of genuine image we used 91 pair of different faces under same lighting. Similarly, to simulate cases of spliced image we used 91 pair of different faces under different lighting.

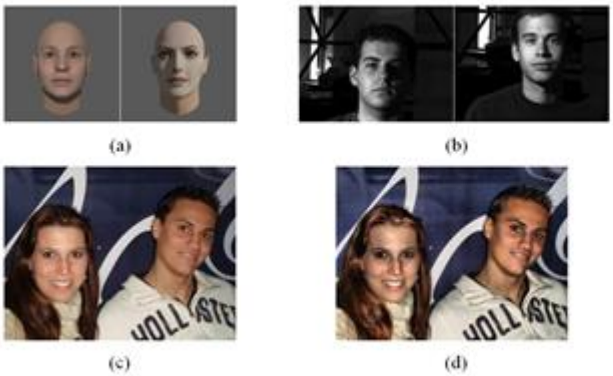

Fig. 3. Examples of images from datasets (a) Syn1 and Syn2, (b) ExtendedYaleB, (c) DSO-1 and post-processed image from DSO-1.

First we performed tests on synthesized faces. Images from Syn1 were compared with images from Syn2. Table I shows the rate of correct detection for tests on Syn1 and Syn2. The values in the table are obtained using thresholds with which the total percentage of errors is minimal. For base FISH distance without normalization threshold is 31, for FISH distance normalized by (7) – 55 and for FISH distance normalized by (8) – 26.

In the next step we performed tests on real faces with controlled lighting. Table II containing results for ExtendedYaleB is similar to the previous one. Optimal thresholds for these tests is 24, 54 and 21.

These tests showed that the developed detector correctly determines the differences in lighting conditions for faces. The next step was to check its performance on real images with and without artificial distortions. For this test we took 95 pristine and 95 spliced images from DSO-1. In Table III the rate of correct detection with optimal thresholds (31, 55 and 28) is presented.

TABLE I. CORRECT DETECTION RATE FOR SYNTHESIZED FACES

Type of FISH descriptor	Correctly detected pristine images	Correctly detected fake images
Base	0.8690	0.9048
Normalized by (7)	0.9286	0.8810
Normalized by (8)	0.9167	0.9643

TABLE II. CORRECT DETECTION RATE FOR REAL FACES WITH CONTROLLED LIGHTING

Type of FISH descriptor		Correctly detected pristine images	Correctly detected fake images
Base		0.9890	0.9560
Normalized by (7)	0.9560	0.9670	
Normalized by (8)	0.9890	0.9670	

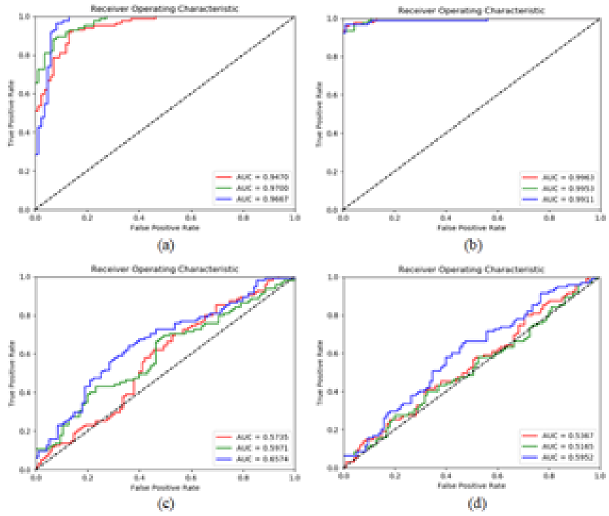

Fig. 4. ROC curves for tests on (a) synthesized faces, (b) real faces with controlled lighting, (c) real images and (d) post-processed real images.

TABLE III. CORRECT DETECTION RATE FOR REAL IMAGES

Type of FISH descriptor	Correctly detected pristine images	Correctly detected fake images
Base	0.5474	0.6105
Normalized by (7)	0.5053	0.6947
Normalized by (8)	0.7158	0.5579

According to results for real images we can see that the presence of several light sources and imperfect surfaces, as well as the high correspondence of spliced faces to lighting conditions of the scene affect the efficiency of the algorithm. Also, we can note that FISH descriptor computed by (8) turned out to be the most effective and it can be used to detect fake images.

However, in order to hide the presence of distortions an attacker can conduct additional post-processing by equalization. Therefore, the effect of this operation on the result of the algorithm was evaluated. An algorithm of contrast-limited adaptive histogram equalization with limit value 0.005 was chosen for this test. It divides the image into squares, in each of which a redistribution of intensity values occurs. After that neighboring squares are combined using bilinear interpolation. As a result of these operations noise may form in the image. To minimize it, the increase in contrast in CLAHE is limited. This operation can hide borders that appear after splicing and reduce the lighting difference of faces in the image. In Table IV we present the correct detection rate for post-processed images, but instead of optimal thresholds we used here the ones we used in previous test for real images without post-processing.

TABLE IV. CORRECT DETECTION RATE FOR POST-PROCESSED REAL IMAGES

Type of FISH descriptor	Correctly detected pristine images	Correctly detected fake images
Base	0.4000	0.6211
Normalized by (7)	0.2105	0.7895
Normalized by (8)	0.5579	0.6000

From ROC curves in Fig. 4 it can be seen that detector performance on post-processed images decreases, but FISH descriptor from (8) still can be used.

V. Conclusion

The developed algorithm correctly determines the difference in lighting conditions of two faces in one image. On real fake images, where this difference is minimal, its effectiveness is not so high, especially if the attacker carried out post-processing. However, the best-performing FISH distance normalized by the average RGB values of pixels of two faces can help to determine the authenticity of the image.

Acknowledgment

This research is sponsored by the Russian Federal Property Fund in the framework of scientific projects № 20-37-70053, 19-07-00138, 19-07-00474.

References

[1] A. Piva, “An overview on image forensics,” ISRN Signal Processing, 496701, 2013.

[2] B. Mahdian and S. Saic, “Using noise inconsistencies for blind image forensics,” Image and Vision Computing, vol. 27, no. 10, pp. 1497-1503, 2009.

[3] B. Li, T. Ng, X. Li, S. Tan and J. Huang, “Revealing the trace of high-quality JPEG compression through quantization noise analysis,” Information Forensics and Security, IEEE Transactions, vol. 10, no. 3, 558-573, 2015.

[4] M. Chen, J. Fridrich, M. Goljan and J. Lukáš, “Determining image origin and integrity using sensor noise,” IEEE Transactions on Information Forensics and Security, vol. 3, no. 1, pp. 74-90, 2008.

[5] H. Yao, S. Wang, Y. Zhao and X. Zhang, “Detecting image forgery using perspective constraints,” IEEE Signal Processing Letters, vol. 19, pp. 123-1126, 2012.

[6] E. Kee, J. F. O’Brien and H. Farid, Exposing photo manipulation with inconsistent shadows, ACM Trans. Graph. 32 (3) (2013) 28:1–28:12.

[7] M. K. Johnson and H. Farid, “Exposing digital forgeries in complex lighting environments,” IEEE Transactions on Information Forensics and Security, vol. 2, no. 3, pp. 450-461, 2007.

[8] M.K. Johnson and H. Farid, “Exposing digital forgeries through specular highlights on the eye,” Proceedings of the 9th International Conference on Information Hiding, vol. 4567, pp. 311-325, 2007.

[9] M. Fanfani, F. Bellavia, M. Iuliani, A. Piva and C. Colombo, “FISH: Face Intensity-Shape Histogram Representationfor Automatic Face Splicing Detection,” Journal of Visual Communication and Image Representation, vol. 63, 102586, 2019.

[10] B. Peng, W. Wang, J. Dong and T. Tan, “Optimized 3d lighting environment estimation for image forgery detection,” IEEE Transactions on Information Forensics and Security, vol. 12, no. 2, pp. 479-494, 2017.

[11] A.S. Georghiades, P.N. Belhumeur and D.J. Kriegman, “From few to many: illumination cone models for face recognition under variable lighting and pose,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 6, pp. 643-660, 2001.

[12] T. Carvalho, C. Riess, E. Angelopoulou, H. Pedrini and A. de Rezende Rocha, “Exposing digital image forgeries by illumination color classification,” IEEE Transactions on Information Forensics and Security, pp. 1182-1194, 2013.