Detecting Video Forgery by Estimating Extrinsic Camera Parameters

Xianglei Hu[1](✉), Jiangqun Ni[1,2], and Runbiao Pan[1]

[1] Sun Yat-Sen University, Xingang Xi Road No. 135,
Guangzhou 510275, People's Republic of China
huxiangl@mail2.sysu.edu.cn, issjqni@mail.sysu.edu.cn

[2] State Key Laboratory of Information Security,
Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093,
People's Republic of China

Abstract. Nowadays, people can easily combine several videos into a fake one by means of matte painting to create visually convincing video contents. This raises the need to verify whether a video content is original or not. In this paper we propose a geometric technique to detect this kind of tampering in video sequences. In this technique, the extrinsic camera parameters, which describe positions and orientations of camera, are estimated from different regions in video frames. A statistical distribution model is then developed to characterize these parameters in tampering-free video and provides evidences of video forgery finally. The efficacy of the proposed method has been demonstrated by experiments on both authentic and tampered videos from websites.

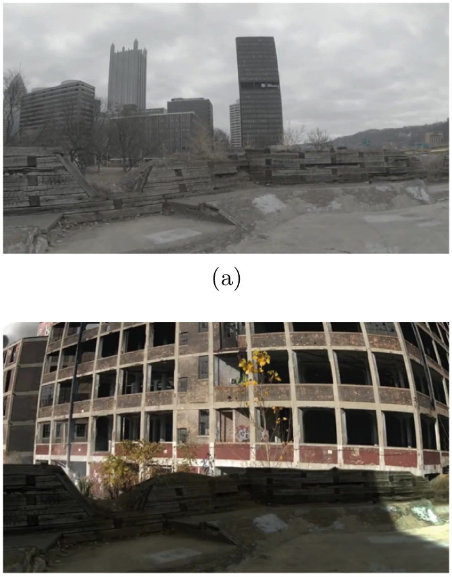
Keywords: Forgery detection · Video forensics · Extrinsic camera parameter

1 Introduction

More and more techniques and software, such as Adobe After Effect and Corel Video Studio, provide to people the convenience of editing and altering videos. Among all the techniques, matte painting is one which can combine several video materials together, and it is widely used in movie effect area. However, by taking advantages of matte paining, people can also make fake videos for evil purposes. Since all the video materials components are real, it is not easy to extract obvious visual clues from the fake video (as in Fig. 1). To tackle this kind of problem, we propose a brand new digital forensic method to detect whether a video is authentic or faked by matte painting.

A lot of work have been done for different kinds of digital video forensics. Milani et al. outlined the video forensic technologies of different kinds of forgeries [1]. Wang and Farid successfully worked out the problem of interlaced and de-interlaced video [2]. Stamm et al. used the fingerprint model to detect the frame deleting/adding operations [3]. Hsu et al. used the temporal noise correlation to detect video forgery, however the model is sensitive to the quantization

Y.-Q. Shi et al. (Eds.): IWDW 2015, LNCS 9569, pp. 28–38, 2016.
DOI: 10.1007/978-3-319-31960-5_3



(a)

(b)

Fig. 1. A true (a) and fake (b) video are shown. The background region is replaced by another video clip and visual clues are hardly seen.

noise [4]. Chen and Fridrich used characteristics of the sensor noise to detect the tampering [5]. However, since lots of effects and recompression have been added to videos during the editing process, these methods can hardly detect the forgery implemented by Chroma key composition. Lighting [6], shadows [7] and reflections [8] are also used for forensics. But these content-based methods do not perform well under poor illumination conditions. The copy-move detecting techniques, such as [9], may not work properly because composites are not from the same source video. Thus, geometric methods are more suitable for the matte painting forensic task. Yao used perspective constraints to detect forgery [10]. Single-view metrology is the theoretical basis of that method, and ideal perspective effects and priori knowledge of objects are used to detect the forgery in images or videos. On the other hand, multi-view metrology based methods mainly focus on ways of detecting forgery by means of planar constraints [11–13]. However, these methods are applicable only when the fake contents are coplanar. To tackle more general matte painting problem in video, we propose a geometric technique using extrinsic camera parameters in this paper. This method implements multi-view metrology to estimate extrinsic camera parameters, and then we focus on investigating the differences of extrinsic parameters estimated from different regions in video frame. We find that regions can be characterized by the extrinsic parameters, and the difference of parameters can help to reveal the matte painting forgery. Experiments shows that our method is robust and efficient, even under the non-coplanar condition.

2 Extrinsic Camera Parameter Estimation

Extrinsic camera parameters are usually introduced to model the position and orientation of cameras. Currently, Structure from Motion (SfM) is one of the most popular methods to estimate extrinsic camera parameters from multi-view images [14,15]. Usually, for simplicity, a camera can be modeled as a pinhole camera. Let $p = [x, y]^T$ denote a 2D point in the image coordinate system. Similarly, $P = [X, Y, Z]^T$ denotes a 3D point in the world coordinate system. $p = [x, y, 1]^T$ and $P = [X, Y, Z, 1]^T$ denote the augmented vectors of them respectively.

In the pinhole camera model, a 3D real world point P and its projection p on the image plane satisfies:

$$sp = \mathbf{K} \left[\mathbf{R}\ \mathbf{t} \right] P \tag{1}$$

where s is a scale factor; $\mathbf{K}$ is the intrinsic camera parameter matrix which carries the information such as the focal length, skewness and principal point of a camera; the extrinsic camera parameters, $\mathbf{t}$ and $\mathbf{R}$, represent the translation and rotation from the world coordinate to the image coordinate system. $\mathbf{t}$ is a 3×1 matrix and $\mathbf{R}$ is a 3×3 matrix:

$$\mathbf{t} = [T_x, T_y, T_z]^T \tag{2}$$

and

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{3}$$

where

$$\begin{aligned}
r_{11} &= \cos\beta\cos\gamma, r_{12} = \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\
r_{13} &= \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\
r_{21} &= \cos\beta\sin\gamma, r_{22} = \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma \\
r_{23} &= \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma \\
r_{31} &= -\sin\beta, r_{32} = \sin\alpha\cos\beta \\
r_{33} &= \cos\alpha\cos\beta
\end{aligned}$$

α, β and γ are Euler angles representing three elementary rotations around $x, y, z-axis$ respectively. In this paper we use the rotation angle vector $\mathbf{r}$ instead of the rotation matrix in later sections:

$$\mathbf{r} = [\alpha, \beta, \gamma]^T. \tag{4}$$

When a camera moves in a scene and takes photos of the same object from different views, it is easy for us to find corresponding points of that same object in these photos. Let p_1 denote a point in image $\mathbf{I}_1$ and its corresponding point p_2 in image $\mathbf{I}_2$. $\mathbf{I}_1$ and $\mathbf{I}_2$ are images of the same object taken from different views. p_1 and p_2 satisfy the fundamental matrix constraint as follows:

$${p_2'}^T \mathbf{F} p_1 = 0 \tag{5}$$

where $\mathbf{F}$ is the fundamental matrix which relates corresponding points in the stereo image pair:

$$\mathbf{F} = \mathbf{K_2}^{-T}\hat{\mathbf{T}}\mathbf{R}\mathbf{K_1}^{-1}, \hat{\mathbf{T}} = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix}.$$

Remarkably, if points in $\mathbf{I}_1$ and $\mathbf{I}_2$ are coplanar in the real world scene, the fundamental matrix constraint will degenerate to the planar constraint which is used in [11–13].

By matching enough corresponding points (at least 8 valid points for each pair) among multi-view images, we can solve the constraint problem and get the fundamental matrix $\mathbf{F}$ [16]. Given $\mathbf{F}$, we can further get $\mathbf{R}$ and $\mathbf{t}$ as well as $\mathbf{K}$. In this way, we can estimate successfully both intrinsic and extrinsic camera parameters. In [17], intrinsic parameters are applied to detect some kinds of video forgery in which the matte painting forgery is not included. Extra information, such as extrinsic parameters, is needed for such kind of forensic, and this paper focuses on how to explore the utility of extrinsic parameters.

3 Proposed Method

Our method is based on utilization of extrinsic camera parameters. Theoretically, in frames of authentic videos, all of the corresponding points should hold the same fundamental matrix constraint (5). Therefore, the same extrinsic camera parameters are supposed to be estimated from corresponding points in a authentic video. If we have extracted different extrinsic camera parameters from different image regions in the same video, it means the video has been tampered. In this way, the forgery in the video can be detected successively.

Steps of the proposed method are arranged as follows. Firstly, we divide each video frame into several different regions with masks. Secondly, we estimate extrinsic parameters from these regions respectively and calculate differences of the parameters. Thirdly, if the threshold is exceeded by the differences between a certain region and all other regions, this region will be considered as a fake one; otherwise, this region will be considered as an authentic one. Figure 2 shows the diagram of our proposed method.

3.1 Estimating Extrinsic Camera Parameters

There are many softwares for estimating the extrinsic camera parameter. Before estimating, we employ the SIFT algorithm to extract feature points [18] and RANSAC algorithm to match them [19]. Then we use the software VisualSFM [20, 21] for estimation and bundle adjustment to refine the result [22].

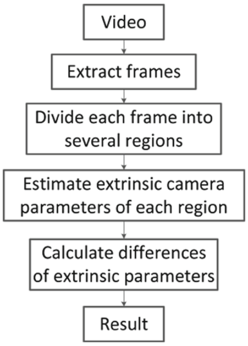

Video
Extract frames
Divide each frame into several regions
Estimate extrinsic camera parameters of each region
Calculate differences of extrinsic parameters
Result

Fig. 2. Diagram of our proposed method

3.2 Detecting Forgeries with Extrinsic Camera Parameters

Even when applying the parameter estimation in a video without any tampering, it is difficult for us to get the exactly same result every time. Many factors, such as mismatched corresponding points, distortion of lens, will lead to the fluctuation of results.

Assuming elements in translation vector (2) and rotation angle vector (4) are independent and identically distributed(i.i.d) respectively, the translational and rotational differences between the estimated and the ground truth values should follow the zero mean Gaussian distribution, i.e.,

$$\mathbf{t}_{est} - \mathbf{t}_{truth} \sim N(\mathbf{0}, {\sigma_t}^2 \mathbf{I}) \tag{6}$$

$$\mathbf{r}_{est} - \mathbf{r}_{truth} \sim N(\mathbf{0}, {\sigma_r}^2 \mathbf{I}) \tag{7}$$

where $\mathbf{t}_{est}$ and $\mathbf{r}_{est}$ denote the estimated translation vector and rotation angle vector respectively, $\mathbf{t}_{truth}$ and $\mathbf{r}_{truth}$ are the ground truth vectors, $\mathbf{I}$ is the unit covariance matrix.

If we divide a video frame into N regions and estimate the extrinsic parameter vectors for each region, we will get the translation vectors $\mathbf{t}_i$ from the ith region and $\mathbf{t}_j$ from the jth region. We define the translational difference between $\mathbf{t}_i$ and $\mathbf{t}_j$ as follows:

$$DT_{ij} = \frac{||\mathbf{t}_i - \mathbf{t}_j||^2}{{\sigma_t}^2} \tag{8}$$

where $i, j = 1, 2, ..., N$ and $i \neq j$; $||.||$ is the L2-norm of vector. Since the square of the L2-norm is equally the sum of squares, and meanwhile all elements of the vector are mutually independent Gaussian random variables, and thus the translational difference DT_{ij} should follow the chi-squared distribution with 3 degrees of freedom (since the vector contains 3 elements):

$$DT_{ij} \sim \chi^2(3). \tag{9}$$

We define the rotational difference DR_{ij} in the same way:

$$DR_{ij} = \frac{||\mathbf{r}_i - \mathbf{r}_j||^2}{{\sigma_r}^2} \tag{10}$$

$$DR_{ij} \sim \chi^2(3). \tag{11}$$

Usually the standard deviation factor is related to the ground truth parameters:

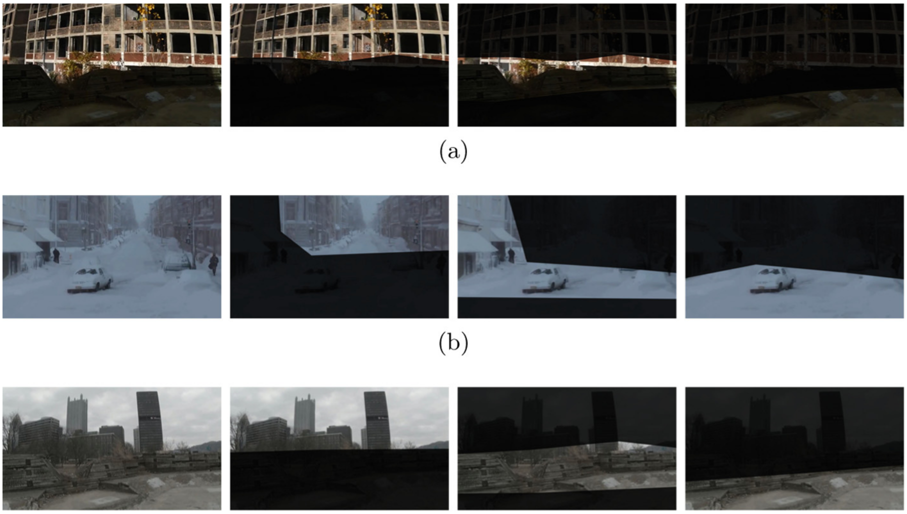
$$\sigma_t = k_t||\mathbf{t}_{truth}|| \tag{12}$$

$$\sigma_r = k_r||\mathbf{r}_{truth}|| \tag{13}$$

where k_t is the total translation factor and k_r is the total rotation factor.

With respect to the ith region, if the mean of DT_{ij} and DR_{ij} from all frames exceed the threshold, we can claim that the ith region is tampered. In this paper, the threshold is set 7.82 which comes from the χ^2 value given the 0.05 p-value of the chi-squared distribution with 3 degrees of freedom, and it indicates that the probability that the object value exceeds 7.82 is less than 0.05.

4 Experiments



(a)

(b)

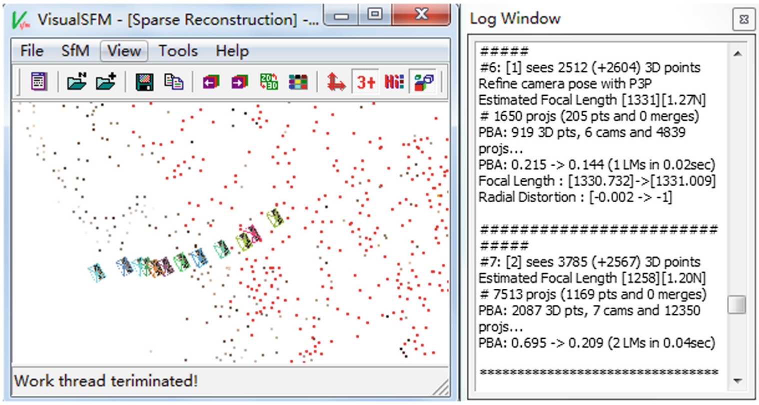
(c)

Fig. 3. Test videos from YouTube. The first column shows the first frames. Column 2 to 4 show the three divided regions. (a) and (b) are tampered videos. (c) is the true version of (a). Column 2 in (a) and (b) show the tampered regions.

4.1 Forensic Model Training

To estimate the total factor k_t and k_r in (12) and (13), we collect more than 50 true video clips, which are either taken by ourselves or downloaded from video-sharing websites. Then, more than 20 frames are extracted from each video. Next, we use Adobe Photoshop's mask tool to generate three new pictures from the original frame. Each new picture contains only one part information of the original frame, while the rest of the new picture contains nothing but black by setting RGB values to 0. The strategy which we run for segmenting frames, is that divide suspicious part from others as much as possible, and in the meantime, make sure each part contains enough feature points to keep VisualSFM work efficiently. So far, we get all triple-segmenting sub-region frame sequences resembling Fig. 3.

To extract the extrinsic camera parameters, the sub-region frame sequences are separately sent to VisualSFM. Afterward, we get extrinsic parameters, the position and orientation, of each sub-frame, as in Fig. 4.



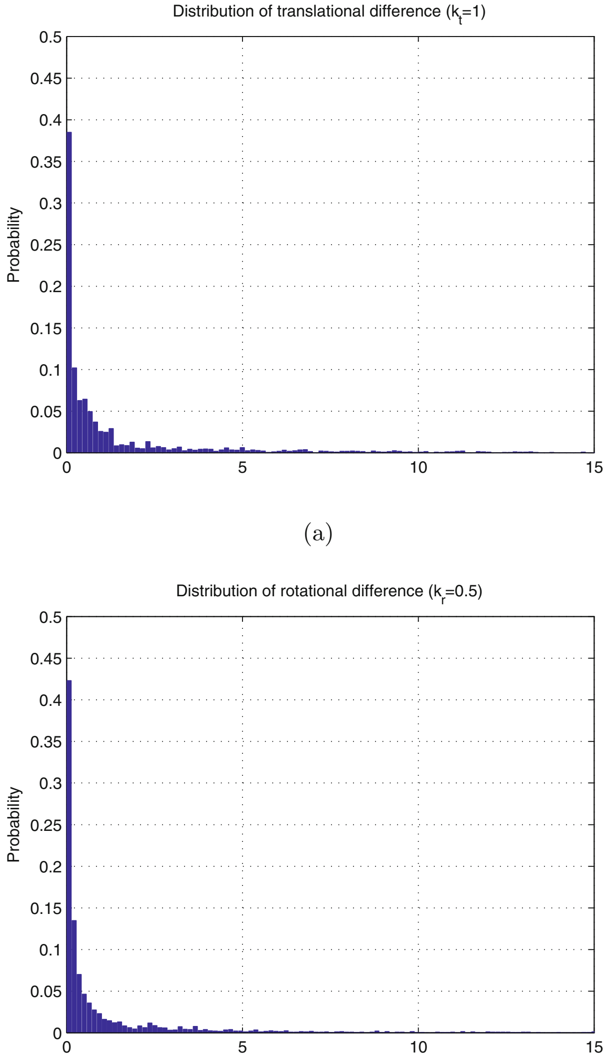
```
Work thread teriminated!
```

```
#####
#6: [1] sees 2512 (+2604) 3D points
Refine camera pose with P3P
Estimated Focal Length [1331][1.27N]
# 1650 projs (205 pts and 0 merges)
PBA: 919 3D pts, 6 cams and 4839
projs...
PBA: 0.215 -> 0.144 (1 LMs in 0.02sec)
Focal Length : [1330.732]->[1331.009]
Radial Distortion : [-0.002 -> -1]

#########################
#####
#7: [2] sees 3785 (+2567) 3D points
Estimated Focal Length [1258][1.20N]
# 7513 projs (1169 pts and 0 merges)
PBA: 2087 3D pts, 7 cams and 12350
projs...
PBA: 0.695 -> 0.209 (2 LMs in 0.04sec)

*********************************
```

Fig. 4. Results provided by VisualSFM, the sequence of rectangles stand for the cameras taking different frames, while the points stand for corresponding feature 3D points from sub-region frames.

However, when we estimate the extrinsic camera parameters, the ground truth are always unknown. Thus, when evaluating our method, we take the mean value $\bar{\mathbf{t}}$ of translation vectors extracted from different regions of all frames as the ground truth translation vector $\mathbf{t}_{truth}$ in (12). And $\bar{\mathbf{r}}$ is taken in the similar way.

After investigating the distribution of translational and rotational difference (as in Fig. 5), we find that 95 % of the difference values are less than 7.82 when $k_t = 1$ and $k_r = 0.5$. Thus, we take $k_t = 1$ and $k_r = 0.5$ in our later experiments.



Distribution of translational difference (k_t=1)

(a)

Distribution of rotational difference (k_r=0.5)

(b)

Fig. 5. Distribution of translational difference and rotational difference. The red dash line shows the χ^2 distribution with 3 degrees of freedom. About 95 % of the difference values are less than 7.82.

4.2 Test of Fake Videos

Then we evaluate our method by examining some video clips obtained from video-sharing websites such as YouTube. We divide each video frame into three regions so that $N = 3$ in (8) and (10), as same as the steps of model training. Figure 3 shows the example. In Fig. 3, (a) and (b) show the tampered videos while (c) shows the authentic version of (a). The first column shows the first frames extracted from the videos. Column 2 to 4 show the three divided regions from top to bottom of the frame. The top regions of both (a) and (b) (as in Column 2) are tampered regions. Here the three regions are simply denoted as Region 1, 2 and 3. The region index pair for calculating translational difference and rotational difference is denoted as (i, j). The result is shown in Table 1.

Table 1. Differences of extrinsic camera parameters for detecting forgery on videos from YouTube

Video	Region pair (i, j)	$D\bar{T}_{ij}$	$D\bar{R}_{ij}$	Predicted tampered region
a	(1, 2)	**12.891**	**15.971**	Region 1
	(1, 3)	**13.636**	**14.131**	
	(2, 3)	3.508	1.412	
b	(1, 2)	**16.500**	**25.381**	Region 1
	(1, 3)	**17.731**	**22.434**	
	(2, 3)	0.583	1.658	
c	(1, 2)	0.078	0.055	None
	(1, 3)	0.592	0.431	
	(2, 3)	0.765	0.447	

In video (a), the whole Region 1 is the suspicious part (the building). The mean value of translational difference and rotational difference are both greater than the thresholds, and thus, Region 1 is predicted to be the tampered region. Region 2 contains a small suspicious part (the building) and has a little great difference of translation. Since most of Region 2 is true, our method predicts that it is authentic as well. Region 3 is totally authentic and has small differences of both translation and rotation.

In video (b), our method can point out the tampered region as well. Video (c) is the authentic version of video (a). The translational difference and rotational difference are both much smaller than the thresholds and no region is predicted fake.

Experiments of other test videos have the similar results. Our proposed method can detect fake regions by taking advantages of extrinsic camera parameters in videos.

5 Conclusion

We proposed a geometric method to detect forgery in video by means of extrinsic camera parameters. For a authentic video, no matter which frame region we use for camera parameter estimation, the estimated extrinsic parameters should not deviate much. Instead of purifying, we try to model the difference of extrinsic parameters in authentic videos so that we can distinguish the fake in a general way. With several real videos, we investigate the differences of extrinsic camera parameters extracted from different regions of the frame. We find that the translational difference and rotational difference follow the chi-squared distribution with 3 degrees of freedom. Then we choose the appropriate threshold for forensics from this distribution. Experiments on videos from video-sharing websites show the efficacy of our method.

Acknowledgment. The authors appreciate the supports received from National Natural Science Foundation of China (No. 61379156 and 60970145), the National Research Foundation for the Doctoral Program of Higher Education of China (No. 20120171110-037) and the Key Program of Natural Science Foundation of Guangdong (No. S2012020011114).

References

1. Milani, S., Fontani, M., Bestagini, P., Barni, M., Piva, A., Tagliasacchi, M., Tubaro, S.: An overview on video forensics. APSIPA Trans. Sig. Inf. Process. **1**, e2 (2012). Cambridge Univ Press, Cambridge
2. Wang, W., Farid, H.: Exposing digital forgeries in interlaced and deinterlaced video. IEEE Trans. Inf. Forensics Secur. **2**(3), 438–449 (2007). IEEE Press, New York
3. Stamm, M.C., Lin, W.S., Liu, K.J.: Temporal forensics and anti-forensics for motion compensated video. IEEE Trans. Inf. Forensics Secur. **7**(4), 1315–1329 (2012). IEEE Press, New York
4. Hsu, C.C., Hung, T.Y., Lin, C.W., Hsu, C.T.: Video forgery detection using correlation of noise residue. In: 2008 IEEE 10th Workshop on In Multimedia Signal Processing, pp. 170–174. IEEE Press, New York(2008)
5. Chen, M., Fridrich, J., Goljan, M., Lukas, J.: Determining image origin and intergrity using sensor noise. IEEE Trans. Inf. Forensics Secur. **3**(1), 74–90 (2008). IEEE Press, New York
6. Johnson, M.K., Farid, H.: Exposing digital forgeries by detecting inconsistencies in lighting. In: Proceedings of the 7th Workshop on Multimedia and Security, pp. 1–10. ACM (2005)
7. Kee, E., O'Brien, J.F., Farid, H.: Exposing photo manipulation with inconsistent shadows. ACM Trans. Graph. **32**(4), 28 (2013). 1C-12. ACM
8. O'Brien, J.F., Farid, H.: Exposing photo manipulation with inconsistent reflections. ACM Trans. Graph. **31**(1), 4 (2012). 1C-11. ACM
9. Wang, W., Farid, H.: Exposing digital forgeries in video by detecting duplication. In: Proceedings of the 9th Workshop on Multimedia and Security, pp. 35–42. ACM (2007)
10. Yao, H., Wang, S.: Detecting image forgery using perspective constraints. Signal Process. Lett. **19**(3), 123–126 (2012). IEEE Press, New York

11. Wang, W., Farid, H.: Detecting Re-projected Video. Proceedings of International Workshop on Information Hiding. Springer, Heidelberg (2008)
12. Conotter, V., Boato, G., Farid, H.: Detecting photo manipulation on signs and billboards. In: 2010 17th IEEE International Conference on Image Processing, pp. 1741–1744. IEEE Press, New York (2010)
13. Zhang, W., Cao, X., Qu, Y., Hou, Y., Zhao, H., Zhang, C.: Detecting and extracting the photo composites using planar homography and graph cut. IEEE Trans. Inf. Forensics Secur. **5**(3), 544–555 (2010). IEEE Press, New York
14. Zhang, Z.: Flexible camera calibration by viewing a plane from unknown orientations. In: Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 1, pp. 666–673 (2010)
15. Nister, D.: An efficient solution to the five-point relative pose problem. IEEE Trans. Pattern Anal. Mach. Intell. **26**(6), 756–770 (2004). IEEE Press, New York
16. Hartley, R.: In defense of the eight-point algorithm. IEEE Trans. Pattern Anal. Mach. Intell. **19**(6), 580–C593 (1997). IEEE Press, New York
17. Johnson, M.K., Farid, H.: Detecting photographic composites of people. In: Proceedings of International Workshop on Digital Watermarking, pp. 19–33. Springer, Heidelberg (2008)
18. Lowe, D.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **2**(66), 91–110 (2004). Springer, Heidelberg
19. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysisand automated cartography. Commun. ACM **24**(6), 381–395 (1981). ACM
20. Wu, C.: Towards linear-time incremental structure from motion. In: 2013 International Conference on 3D Vision-3DV 2013, pp. 127–134 (2013)
21. Wu, C.: VisualSFM: A Visual Structure from Motion System. http://ccwu.me/vsfm/
22. Wu, C., Agarwal, S., Curless, B., Seitz, S.M.: Multicore bundle adjustment. In: IEEE Conference on Computer Vision and Pattern Recognition, pp: 3057–3064. IEEE Press, New York (2011)