symmetry

MDPI

Article

Forgery Detection and Localization of Modifications at the Pixel Level

Sahib Khan [1,*], Khalil Khan [2], Farman Ali [3] and Kyung-Sup Kwak [4,*]

[1] Department of Electronics and Telecommunications, Politecnico di Torino, 10129 Torino, Italy
[2] Department of Electrical Engineering, University of Azad Jammu and Kashmir, Muzaffarabad 13100, Pakistan; khalil.khan@ajku.edu.pk
[3] Department of Software, Sejong University, Seoul 05006, Korea; farmankanju@sejong.ac.kr
[4] Department of Information and Communication Engineering, Inha University, Incheon 22212, Korea
* Correspondence: sahib.khan@polito.it (S.K.); kskwak@inha.ac.kr (K.-S.K.)

Received: 21 December 2019; Accepted: 6 January 2020; Published: 9 January 2020

Abstract: In this paper, we present a new technique of image forgery detection. The proposed technique uses digital signatures embedded in the least significant bits of the selected pixels of each row and column. The process maintains a symmetry in the use of pixels for computing and hiding the digital signatures. Each row and column of the image symmetrically contributes to both processes, with the number of pixels per row or column used for computing the signature, and the pixels used for embedding are not equal and are asymmetric. The pixels in each row and column of an image are divided into two groups. One group contains pixels of a row or column used in the calculation of digital signatures, and the second group of pixels is used for embedding the digital signatures of the respective row or column. The digital signatures are computed using the hash algorithm, e.g., message digest five (MD5). The least significant bits substitution technique is used for embedding the computed digital signature in the least significant bits of the selected pixels of the corresponding row or column. The proposed technique can successfully detect the modification made in an image. The technique detects pixel level modification in a single or multiple pixels.

Keywords: forgery detection; message digest 5; LSB substitution; accuracy

1. Introduction

Currently, with the growth of technology, high resolution digital cameras are available at reasonable prices. Along with standalone cameras, digital cameras are available on each smartphone. This has made the saving of important events and memory very easy. Selfie capturing has become fashionable. With all these uses, digital images can also play the role of pieces of evidence. The images of crime scenes can be presented in a court of law as proof.

Along with the high resolution cameras, various image editing tools and software have been developed to enhance the quality of images and other useful purposes. However, at the same time, these tools have raised the chances of the misuse of digital images. The image editing tools can be used to forge image contents, which can portray false information. The forged image can be used to affect the image of a person, harass a person, and as false evidence in a court of law. The modification of digital images has become of concern to people (e.g., fake and modified images of well known personalities and big names of society), societies, journalism, technical research publication, etc. [1]. This raises the question of how much the contents of images can be trusted.

Images can be modified by splicing, re-sampling, removing and adding a part, etc. The changes made may be detectable to the human visual system (HVS), and the modified contents can be identified with the naked eye. In such scenarios, the trustworthiness of the image contents can be decided by

merely observing the modified images. However, the manipulations may affect human life, even more severely in the current era than in the past. Images are modified with the help of advanced tools in such a manner that the modifications are imperceptible to the HVS, and it is difficult, and almost impossible, to detect the manipulation with the naked eye [2]. Such variations give birth to severe vulnerabilities and risk the integrity of the digital images. It is important to how much the contents of an image can be trusted. The authentication of images and detection of the manipulations, if made, in digital images will help to prevent the misuse of digital images and eliminate risks. Only trusted and authentic images are presented in a court of law, especially when the digital contents are produced as legal evidence in front of judges [3].

Therefore, it is important to validate image contents and detect and locate the forged part of the image. Efficient forgery detection can successfully distinguish between the authentic and manipulated images and find the changes made. Due to a variety of applications and public interest, forgery detection is an area of great interest to forensic experts. The image authentication and forgery detection techniques have been classified into two categories, i.e., active techniques and passive techniques. Embedded watermarks or signatures are used in the active techniques, while in passive techniques, no extra information is used. The active techniques have limited applications [4], due to the non-availability of these techniques in all image acquisition devices. Active authentication techniques are further classified into two types: digital signature and digital watermarking [1]. Passive techniques have great application in image processing [5–10]. Passive techniques use the statistics of images for forgery detection and content authentication. Each image acquisition device leaves some non-modifiable, noise-like signal, which can be used to detect the forgery. The passive techniques are classified as pixel based techniques, i.e., copy-move [11,12], image splicing [13], and image retouching [14–22], format based techniques [23–26], camera based techniques [27,28], physical based techniques [29–35], and geometric based techniques [36,37]. These techniques are computationally expensive and time consuming and are not commonly used.

The proposed work is an effort toward image forgery detection at the pixel level. The technique uses a digital signature embedded in the selected pixels row- and column-wise. The digital signatures are computed using the MD5 [38] algorithm and are embedded in LSBs of selected pixels using the LSB substitution method [39,40].

The remainder of the paper is organized into four sections. Section 2 presents the implementation of the proposed framework. The experimental results and analysis based on the results are presented in Section 3. A comparison of the proposed technique with other image forgery detection techniques is given in Section 4. Section 5 concludes the discussion.

2. Proposed Forgery Detection Techniques

The image forgery detection was also addressed in [41,42]. The work in [41] detected image forgery at the row level; however, it failed to detect any modification when a complete row or rows were truncated from an image. While the method presented in [42] detected image forgery at the column level, the algorithm failed to detect a complete column's or columns' truncation. Moreover, the techniques in [41,42] designated a complete row or column as forged, even if a single pixel was modified in a row or column of an image, respectively. The algorithms located the forged rows or columns instead of forged pixels.

The limitations present in [41,42] are addressed in the proposed technique. The proposed framework detects image forgery at the pixel level. The algorithm is also capable of detecting rows' or columns' truncation. It marks only the manipulated pixel or group of pixels as forged. The proposed technique divides the pixels of each row and column into two parts. One group of pixels is used in the computation of the digital signature, while the second group of pixels is used for embedding the digital signatures. Hence, the number of pixels used in both processes is asymmetrical. Digital signatures are calculated for each row and column, and the signatures are embedded in the respective rows and columns. On the other side, to authenticate image contents and detect any possible modification introduced to the

image, the digital signature for each row and column is computed similarly by using the selected pixels. The embedded digital signatures are retrieved from the LSBs of the selected pixels used for embedding. Both the computed and retrieved signatures of each row and columns are compared. If the signatures match each other, the row or column is declared as an authentic one. Otherwise, the row or column for which the computed and retrieved signatures do not match each other is considered unauthentic. As each pixel is a part of a row and a column, if a pixel is modified, the corresponding rows and columns will be labeled as unauthentic. Therefore, the forged pixel is located at the point of intersection of the forged row and column. Hence, the proposed framework successfully identifies and locates the forged pixels.

Let us consider the same image of size $N \times M$, where N is the number of rows and M is the number of columns. D pixels of each row and column are used to hide the digital signature. D depends on the size of the digital signature in bits and the number of bits embedded per pixel. For example, in the case of MD5, the size of a digital signature is 128 bits, and if four bits per pixels are hidden in the LSBs of the pixels, then D must be 32, to accommodate the signature. The remaining $N - D$ pixels of a row and $M - D$ pixels of a column are used for digital signature computation using the hash algorithm. Hence, each of the processes maintains the symmetry in the use of the number of pixels per row and column and uses an equal number of pixels, while computing the digital signature for each row or column. Similarly, the equal number of pixels per rows or columns is used to embed the digital signature in the corresponding rows or columns.

The processes of digital signatures' computation and embedding the signatures are given in Figure 1. The image portion of size $(N - D) \times (M - D)$, as indicated by the black rectangle, shows part of the selected pixels used for row-wise and column-wise digital signature computation. Pixels highlighted by the orange rectangle show the selected pixels of rows and columns used to embed the digital signature computed for the corresponding rows and columns. The pixels neither used in digital signature computation nor signature embedding are left unaffected or used to authenticate the pixels having embedded signatures, as indicated by the blue rectangle in Figure 1. For this, the pixels having the embedded signature of the first row and first column are collectively used to compute a signature, and the signature is embedded in the first column of the small $D \times D$ size portion. Similarly, the second row and second column of the pixel with embedded signatures are processed for signature calculation, and the signature is embedded in the second column of the $D \times D$ sized portion. The process is applied to all pixels of the rows and columns having an embedded signature, and the signatures are embedded in the corresponding column of the $D \times D$ portion of the image.

Figure 1 shows signature computation for each row and column and embedding of the signatures in the respective row or column. The processed image is then transmitted, saved, or shared. The parties interested in the contents of the image will need to check the authenticity of the contents of the images and will try to detect the forged pixels, if any. To detect the possible forged pixels, the receiver will process the image by dividing the image pixels into three parts. The pixels used for calculating digital signatures, the pixels used for embedding the signatures, and the part used to embed the signatures are computed from the pixels with embedded digital signatures. Digital signatures are computed for each row and column using the selected pixels of each row and column. Then, for each and column, digital signatures are retrieved from the selected pixels of the respective row and column, by reading the LSBs of the selected pixels. The computed and retrieved digital signatures are compared with each other for each row and column. The pixel for which a match in digital signatures is found for the respective row and column is declared authentic. If the signatures of the respective row and column do not match the retrieved signatures, the pixel is declared forged. Hence, forged pixels are detected and located. The process of forgery detection and localization is presented in Figure 2.

To detect a forged pixel, we have two pairs of signatures: one pair of signatures is obtained from the row, and another pair of signatures is obtained from the column. If both pairs are similar, then the pixel is said to be authentic. However, there exists a possibility that one pair of signatures may match while another pair of signatures may fail. In such a scenario, the $D \times D$ portion of pixel is used to find

whether a given part with a hidden signature is affected or not. If a group of pixels with an embedded digital signature is found modified, then the pixel is declared authentic. This further strengthens the claim of the proposed algorithm and avoids any possible false forgery detection.

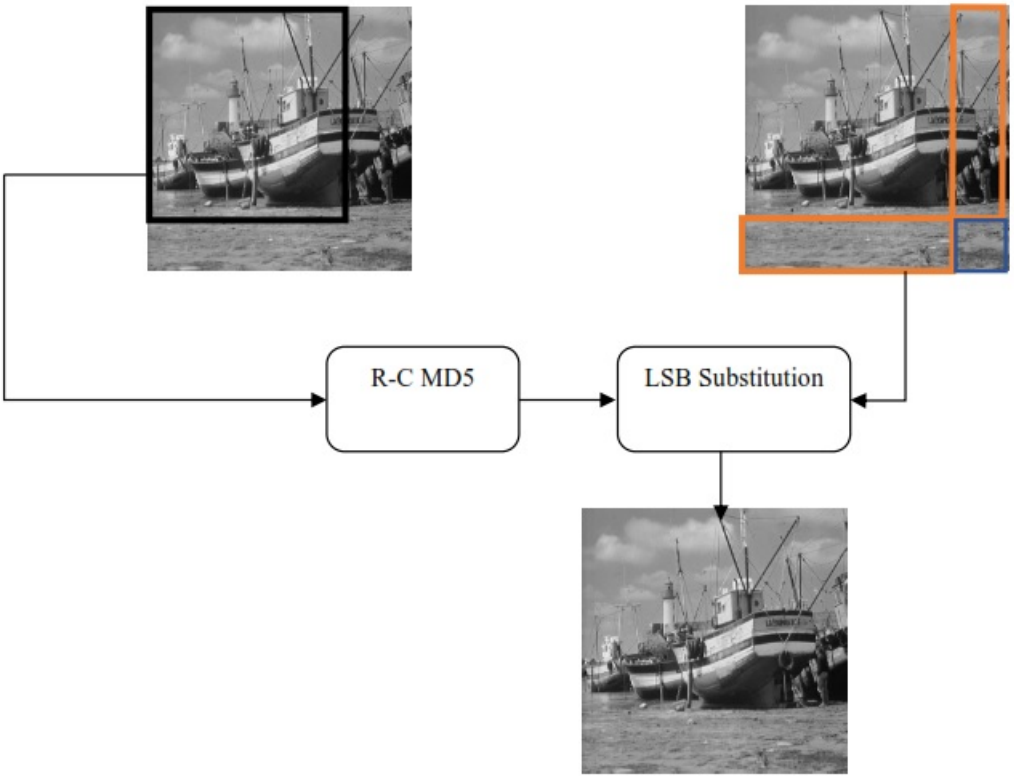

Figure 1. The operation of the proposed technique at the source side.

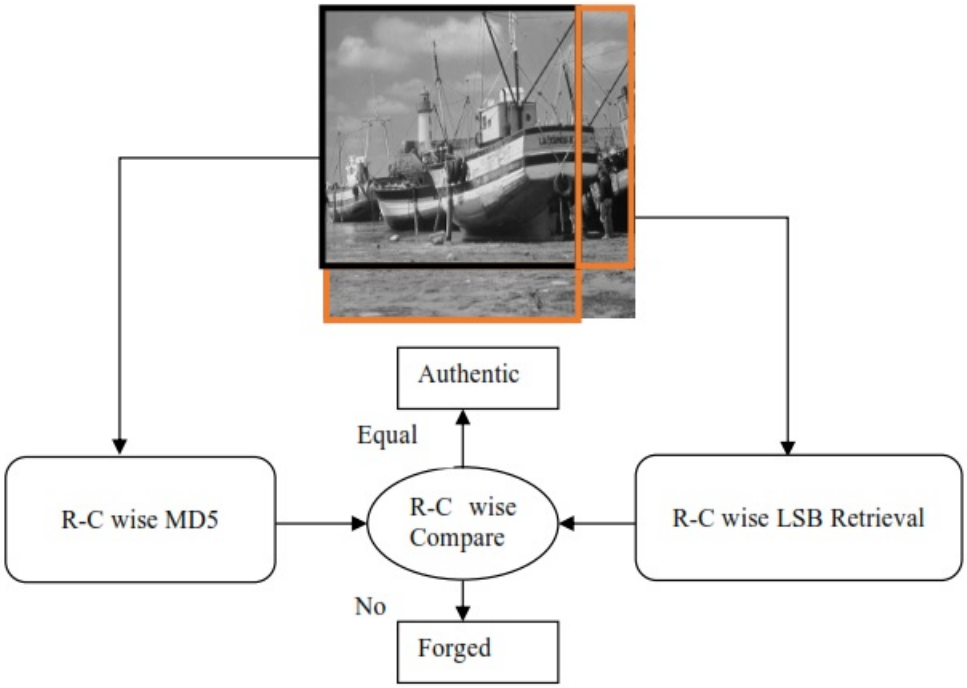

Figure 2. Operation of the proposed technique at receiver side for forgery detection.

3. Experimental Results and Analysis

This section presents a detailed analysis of the proposed technique by performing different experiments. The proposed method computes the digital signature using selected pixels row-wise and column-wise, as discussed in Section 2. The digital signatures are then embedded in the LSBs of the selected pixels of the corresponding rows and columns. There exist various hash algorithms to compute digital signatures. Similarly, several embedding techniques exit in the literature that can be used to embed the signatures. Here, we use the MD5 algorithm to compute the digital signature of 128 bits, and the 4LSB substitution technique is used for embedding four bits per selected pixel. As the digital signature is 128 bits long, we need 32 pixels, i.e., $D = 32$, embedding the full signature using four bits per pixel. Hence, the MD5 algorithm generates a 128 bit signature for each column and row, and 32 selected pixels of each column and row are used for signature embedding. After hiding the digital signatures, we obtain a final image that is stored, transmitted, or shared with the intended user or is made public.

Investigating the authenticity of the contents of the image and detecting any possible modification introduced in the image, at the receiver side, the digital signatures, each of 128 bits, are recalculated from the selected pixels of each row and column using the MD5 algorithm. The embedded digital signatures, each of 128 bits, are retrieved by reading the four LSBs of the selected pixels of each row and column. The retrieved and recalculated digital signatures of each row and column are compared with each other. The forged pixels are identified following the procedure discussed in Section 2.

For analysis, the proposed framework is applied to the image shown in Figure 3a. The image in Figure 3a is processed accordingly for signatures' computation and embedding, as explained in Figure 1. At the end of the embedding process, we obtain a digital image with embedded digital signatures, as shown in Figure 3b.

(**a**) (**b**)

Figure 3. The proposed technique. (**a**) Original image and (**b**) resulting image with embedded digital signatures.

The image with embedded digital signatures is forged by introducing different modifications, e.g., multiple pixels' modification in multiple rows and columns, a block of pixels' modification, single-pixel modification in multiple columns, single-pixel modification in multiple rows, multiple bits' modification in a single pixel, and single bit modification in a single pixel. The images obtained after introducing modification are shown in Figure 4a–f. The images in Figure 4 are then used for forgery detection using the procedure presented in Figure 2.

Figure 4. Modified images. (**a**) Forging different pixels in various rows and columns, (**b**) block of pixels forged, (**c**) single pixel forged in various columns, (**d**) single pixel forged in various rows, (**e**) single pixel subjected to multiple bits' modification, and (**f**) single bit modification in a single pixel.

Each modified image is processed for forgery detection using the proposed technique. It has been observed from the experiments that the proposed technique successfully detects the forged pixels in various scenarios presented in Figure 4. The detected forged pixels are converted into black color, and the forged part of each image is highlighted with the red circle in each image. The experimental results are shown in Figure 5.

Figure 5. Forgery detection in images using the proposed technique. (**a**) Alteration detected in multiple pixels in different rows, (**b**) manipulated block of pixels in an image, (**c**) changes detected in pixels in multiple columns, (**d**) changes detected in pixels in multiple rows, (**e**) different bits' manipulation in one pixel detected, and (**f**) one LSB manipulation in one pixel detected.

4. Comparison

This section presents a comparison of the proposed technique with previous techniques. The comparison is made in terms of true positives (TP), true negatives (TN), and accuracy. The values calculated are listed in Table 1. The results demonstrated the comparison of the proposed techniques with the methods of Lyu and Farids [43], Shi et al. [44], Zou et al. [45], Rad and Wang [46], and Kashyap et al. [47].

The results showed that among all the previous techniques mentioned in Table 1, Kashyap et al.'s technique had the highest detection accuracy of 81.50%, while the techniques Khan et al. [41] and Khan et al. [42] demonstrated detection accuracy equal to 95%. The proposed technique resulted in an

accuracy of 97%. Therefore, it can be concluded that the proposed methods were more powerful than other previous techniques to detect manipulations in digital images.

Table 1. Comparison of the proposed technique with the previous image forgery detection techniques.

	Evaluation Metrics		
Technique	**TP (%)**	**TN (%)**	**Accuracy (%)**
Lyu and Farids	78.20	69.39	73.75
Shi et al.	75.55	76.02	75.78
Zou et al.	77.40	75.07	76.21
Rad et al.	80.11	77.61	78.80
Kashyaop et al.	83.33	76.0	81.50
Khan et al. [41]	96.51	95.78	95.01
Khan et al. [42]	96.51	95.78	95.01
Proposed technique	97.1	96.93	97.02

5. Conclusions

The proposed forgery detection and localization technique was a powerful framework to detect any possible modification introduced to an image. The method successfully detected a single pixel, even a single bit modification. The technique was tested against various types of modifications, and the technique resulted in high accuracy for different scenarios. The comparative analysis showed that the proposed work performed better than the previous techniques. The technique could be used for image forgery detection at any level with great confidence. The proposed algorithm could be used in many applications, e.g., authenticating the contents of images presented in a court of law, and could detect the false presented information or part of the information. It could be used to control the sharing of wrong and misleading information on social media. It could be used in other applications, e.g., assuring data integrity, copyright protection, social networking, online shopping, etc.

Along with the strength of the algorithm, it also had some limitations. For example, it would mark the complete image as forged if the part of pixels having embedded signatures was modified. This would possibly mark a lossy compressed image as a forged image, e.g., in the case of JPEG compression.

Author Contributions: All authors equally contributed in this research work. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation (NRF) of Korea-Grant funded by the Korean Government (Ministry of Science and ICT) NRF-2020R1A2B5B02002478).

Acknowledgments: We are thankful to NRF for providing the funding. This research was supported by NRF of Korea-Grant funded by the Korean Government (Ministry of Science and ICT) NRF-2020R1A2B5B02002478).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Farid, H. Image forgery detection. *IEEE Signal Process. Mag.* **2009**, *26*, 16–25. [CrossRef]
2. Blythe, P.; Fridrich, J. Secure digital camera. In Proceedings of the Digital Forensic Research Workshop, Baltimore, MD, USA, 11–13 August 2004.
3. Cox, I.; Miller, M.L.; Bloom, J.A. *Digital Watermarking*; Morgan Kaufmann: San Mateo, CA, USA, 2001.
4. Redi, J.A.; Taktak, W.; Dugelay, J.L. Digital image forensics: A booklet for beginners. *Multimed. Tool Appl.* **2011**, *51*, 133–162. [CrossRef]
5. Wang, J.; Liu, G.; Zhang, Z.; Dai, Y.; Wang, Z. Fast and robust forensics for image region-duplication forgery. *Acta Autom. Sinica* **2009**, *35*, 1488–1495. [CrossRef]
6. Ansari, M.D.; Ghrera, S.P.; Tyagi, V. Pixel-based image forgery detection: A review. *IETE J. Educ.* **2014**, *55*, 40–46. [CrossRef]

7. Khan, S.; Bianchi, T. Reduced Complexity Image Clustering Based on Camera Fingerprints. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 2682–2688.
8. Fridrich, J. Robust bit extraction from images. In Proceedings of the IEEE International Conference on Multimedia Computing and Systems, Florence, Italy, 7–11 June 1999; pp. 536–540.
9. Zhao, Z.; Li, J.; Li, S.; Wang, S. Detecting digital image splicing in chroma spaces. In *International Workshop on Digital Watermarking*; Springer: Berlin, Germany, 2010; pp. 12–22.
10. Doke, K.K.; Patil, S.M. Digital signature scheme for image. *Int. J. Comput. Appl.* **2012**, *49*, 1–6.
11. Bravo-Solorio, S.; Nandi, A.K. Automated detection and localisation of duplicated regions affected by reflection, rotation and scaling in image forensics. *Signal Process.* **2011**, *91*, 1759–1770. [CrossRef]
12. Pun, C.M.; Yuan, X.C.; Bi, X.L. Image forgery detection using adaptive over segmentation and feature point matching. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 1705–1716.
13. Moghaddasi, Z.; Jalab, H.A.; Noor, R.M. SVD-based image splicing detection. In Proceedings of the 6th International Conference on Information Technology and Multimedia, Putrajaya, Malaysia, 18–20 November 2014; pp. 27–30.
14. Boato, G.; Natale, F.G.; Zontone, P. How digital forensics may help assessing the perceptual impact of image formation and manipulation. In Proceedings of the Fifth International Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale, AZ, USA, 13–15 January 2010.
15. Avcibas, I.; Bayram, S.; Memon, N.; Ramkumar, M.; Sankur, B. A classifier design for detecting image manipulations. In Proceedings of the 2004 International Conference on Image Processing, ICIP '04, Singapore, 24–27 October 2004; pp. 2645–2648.
16. Stamm, M.C.; Liu, K.R. Blind forensics of contrast enhancement in digital images. In Proceedings of the 2008 15th IEEE International Conference on Image Processing, San Diego, CA, USA, 12–15 October 2008; pp. 3112–3115.
17. Stamm, M.C.; Liu, K.R. Forensic estimation and reconstruction of a contrast enhancement mapping. In Proceedings of the 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, Dallas, TX, USA, 14–19 March 2010; pp. 1698–1701.
18. Stamm, M.C.; Liu, K.R. Forensic detection of image manipulation using statistical intrinsic fingerprints. *IEEE Trans. Inf. Forens. Secur.* **2010**, *5*, 492–506. [CrossRef]
19. Cao, G.; Zhao, Y.; Ni, R. Forensic estimation of gamma correction in digital images. In Proceedings of the 2010 IEEE International Conference on Image Processing 2010, Hong Kong, China, 26–29 September 2010; pp. 2097–2100.
20. Li, X.F.; Shen, X.J.; Chen, H.P. Blind identification algorithm for retouched images based on Bi-Laplacian. *J. Comput. Appl.* **2011**, *31*, 239–242. [CrossRef]
21. Cao, G.; Zhao, Y.; Ni, R.; Li, X. Contrast enhancement-based forensics in digital images. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 515–525. [CrossRef]
22. Chierchia, G.; Poggi, G.; Sansone, C.; Verdoliva, L. A Bayesian-MRF approach for PRNU-based image forgery detection. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 554–567. [CrossRef]
23. Fan, Z.; de Queiroz, R. Maximum likelihood estimation of JPEG quantization table in the identification of bitmap compression history. In Proceedings of the International Conference on Image Processing Proceedings, Vancouver, BC, Canada, 10–13 September 2000; pp. 948–951.
24. Fan, Z.; De Queiroz, R.L. Identification of bitmap compression history: JPEG detection and quantizer estimation. *IEEE Trans. Image Process.* **2003**, *12*, 230–235. [PubMed]
25. Lukáš, J.; Fridrich, J. Estimation of primary quantization matrix in double compressed JPEG images. In Proceedings of the Digital Forensic Research Workshop, Cleveland, OH, USA, 6–8 August 2003; pp. 5–8.
26. Popescu, A.C. Statistical Tools for Digital Image Forensics. Ph.D. Thesis, Dartmouth College, Hanover, NH, USA, 2004.
27. Popescu, A.C.; Farid, H. Exposing digital forgeries in color filter array interpolated images. *IEEE Trans. Signal Process.* **2005**, *53*, 3948–3959. [CrossRef]
28. Lukas, J.; Fridrich, J.; Goljan, M. Digital camera identification from sensor pattern noise. *IEEE Trans. Inf. Forensics Secur.* **2006**, *1*, 205–214. [CrossRef]
29. Johnson, M.K.; Farid, H. Exposing digital forgeries by detecting inconsistencies in lighting. In Proceedings of the 7th Workshop on Multimedia and Security, New York, NY, USA, 1–2 August 2005; pp. 1–10.

30. Johnson, M.K.; Farid, H. Exposing digital forgeries in complex lighting environments. *IEEE Trans. Inf. Forensics Secur.* **2007**, *2*, 450–461. [CrossRef]
31. Chen, H.; Shen, X.; Lv, Y. Blind identification method for authenticity of infinite light source images. In Proceedings of the 2010 Fifth International Conference on Frontier of Computer Science and Technology, Changchun, China, 18–22 August 2010; pp. 131–135.
32. Kee, E.; Farid, H. Exposing digital forgeries from 3-D lighting environments. In Proceedings of the 2010 IEEE International Workshop on Information Forensics and Security, Seattle, WA, USA, 12–15 December 2010; pp. 1–6.
33. Lv, Y.; Shen, X.; Chen, H. An improved image blind identification based on inconsistency in light source direction. *J. Supercomput.* **2011**, *58*, 50–67. [CrossRef]
34. Fan, W.; Wang, K.; Cayre, F.; Xiong, Z. 3D lighting-based image forgery detection using shape-from-shading. In Proceedings of the European Signal Processing Conference, Bucharest, Romania, 27–31 August 2012; pp. 1777–1781.
35. De Carvalho, T.J.; Riess, C.; Angelopoulou, E.; Pedrini, H.; de Rezende Rocha, A. Exposing digital image forgeries by illumination color classification. *IEEE Trans. Inf. Forensics Secur.* **2013**, *8*, 1182–1194. [CrossRef]
36. Johnson, M.K.; Farid, H. Detecting Photographic Composites of People. In *International Workshop on Digital Watermarking*; Springer: Berlin, Germany, 2007; pp. 19–33.
37. Ng, T.T.; Chang, S.F.; Hsu, J.; Xie, L.; Tsui, M.P. Physics-motivated features for distinguishing photographic images and computer graphics. In Proceedings of the ACM International Conference on Multimedia, Singapore, 6–12 November 2005; pp. 239–248.
38. Wahid, M.; Ahmad, N.; Zafar, M.H.; Khan, S. On combining MD5 for image authentication using LSB substitution in selected pixels. In Proceedings of the 2018 International Conference on Engineering and Emerging Technologies (ICEET), Lahore, Pakistan, 22–23 February 2018; pp. 1–6.
39. Khan, S.; Yousaf, M.H.; Akram, J. Implementation of Variable Least Significant Bits Stegnography using DDDB Algorithm. *Int. J. Comput. Sci. Issues (IJCSI)* **2011**, *8*, 101–109.
40. Khan, S.; Ahmad, N.; Ismail, M.; Minallah, N.; Khan, T. A secure true edge based 4 least significant bits steganography. In Proceedings of the 2015 International Conference on Emerging Technologies (ICET), Peshawar, Pakistan, 19–20 December 2015; pp. 1–4.
41. Khan, S.; Wahid, M.; Irfan, M.A.; Naeem, M.; Gul, A.; Zafar, M.H. Row Level Image Forgery Detection Technique using Embedded Digital Signature. *J. Eng. Appl. Sci.* **2018**, *37*, 1–6.
42. Khan, S.; Wahid, M.; Khan, T.; Ahmad, N.; Zafar, M.H. Column Level Image Authentication Technique using Hidden Digital Signatures. In Proceedings of the 2018 24th International Conference on Automation and Computing (ICAC), Newcastle upon Tyne, UK, 6–7 September 2018; pp. 1–6.
43. Lyu, S.; Farid, H. Detecting hidden messages using higher-order statistics and support vector machines. In *Information Hiding*; Springer: Berlin/Heidelberg, Germany, 2002; pp. 340–354.
44. Shi, Y.Q.; Xuan, G.; Zou, D.; Gao, J.; Yang, C. Steganalysis based on moments of characteristic functions using wavelet decomposition, prediction-error image, and neural network. In Proceedings of the International Conference on Multimedia and Expo, Amsterdam, The Netherlands, 6 July 2005; pp. 269–272.
45. Zou, D.; Shi, Y.Q.; Su, W.; Xuan, G. Steganalysis based on Markov model of thresholded prediction-error image. In Proceedings of the 2006 IEEE International Conference on Multimedia and Expo, Toronto, ON, Canada, 9–12 July 2006; pp. 1365–1368.
46. Rad, R.M.; Wong, K. Digital image forgery detection by edge analysis. In Proceedings of the 2015 IEEE International Conference on Consumer Electronics-Taiwan, Taipei, Taiwan, 6–8 June 2015; pp. 19–20.
47. Kashyap, A.; Parmar, R.S.; Suresh, B.; Agarwal, M.; Gupta, H. Detection of digital image forgery using wavelet decomposition and outline analysis. In Proceedings of the 2016 International Conference on Signal Processing and Communication (ICSC), Noida, India, 26–28 December 2016; pp. 187–190.