Automation of JPEG Ghost Detection using Graph Based Segmentation

Archana V Mire[1], Dr S B Dhok[2], Dr P D Porey[1], Dr N J Mistry[1]
1 Dept of COED,SVNIT, Surat,
2 Dept of ECED VNIT, Nagpur

ABSTRACT:

As JPEG is widespread de facto image format, Most of the images available in computer systems, electronic devices and in the Web are in JPEG format hence forgery detection techniques in JPEG may explore most of the forgeries. As JPEG works on 8 by 8 block cosine transform most of the tampering correlation inherited by tampered image may get destroyed, making forgery detection difficult. In fact it is common practice followed by forger to hide traces of resampling & splicing. However the tampered region usually has a different JPEG compression history than the authentic region. JPEG ghost detection techniques makes use of artefacts introduced in image due to 8 by 8 block DCT transform. It identifies forgery by searching ghost which appears in resultant difference image after subtracting it from its various recompressed version at different quality levels. As number of difference images become very large it becomes difficult for human being to scan these large no of difference image. Hence in this paper we have proposed a technique which will automate ghost detection in image..

Keywords: JPEG, AJPEG, DJPEG, JPEG ghost, Segmentation, Automation, Difference image

I. INTRODUCTION

With the availability of high quality pirated photo editing software novice users are also able to create convincing image forgery creating big problems for authenticity of digital images. There are various techniques for detection of splicing in image based on inconsistencies of resampling[1][2][3], CFA interpolation[4][5], motion blurr[6]**,** geometric property[7][8][9][10], chromatic aberration[11] & various survey paper [12][13][14] compared these techniques. But these techniques are very subjective to a specific type of forgery detection & performance degrades tremendously with post processing operations & compression. On contrary JPEG intrinsic fingerprint based techniques are more robust to compression. Original image on which forgery is created may be compressed or uncompressed similarly area pasted may belong to compressed or uncompressed image. Since both posses different compression history JPEG forgery detection techniques try to identify difference in compression history which may be in form of DCT block alignment or primary quantization table used. Farid's Ghost detection [15] approach is also based on these blocking artifacts introduced during recomression but it requires lot of human interaction. In this paper we have tried to automate identification of JPEG Ghost & thus avoided manual scanning of difference images. Before stating our automation algorithm we will briefly discuss types of JPEG forgery detection techniques & Farid's Ghost mechanism. JPEG based forgery detection can be broadly classified as below.

1.1 ADJPEG forgery detection

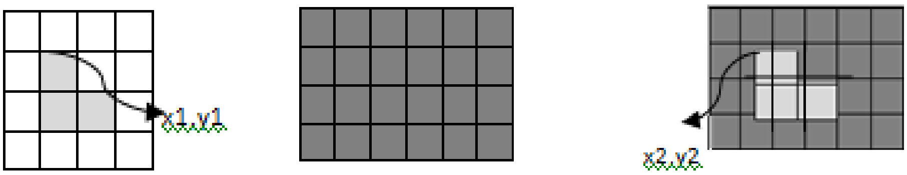
These techniques are dependent on the assumption that JPEG grid used in first compression & second compression shown in figure 1 is exactly aligned with each other satisfying equation (1), (2).

$$cx= (x1 \bmod 8)-(x2 \bmod 8) =0 \qquad \text{(Equ. 1)}$$

$$cy= (y1 \bmod 8)-(y2 \bmod 8) =0 \qquad \text{(Equ. 2)}$$

These techniques mostly identify double compression but not necessarily double compression mean forgery. These techniques can be applied to different objects in the image after segmentation & forgery can be traced out by objects having differing no of the compression history (objects having a single compression history are real & objects having a double compression history are forged).

Farid[16] analyzed ADPJPEG compression with 1D discretely sampled signal. They found that periodicity of the artifacts gets introduced into the histograms of double quantized signal which can be identified as spikes in the Fourier domain & gave complete theoretical proof of this phenomenon which can be used to identify forgery.



a.Source image b.Destination image c.Composite image

Figure 1. Example of JPEG Image forgery

1.2 NADJPEG forgery detection

These techniques are dependent on the assumption that JPEG grid used in first compression & second compression as shown in figure 1 is not aligned with each other satisfying equation (3), (4).

$$cx= (x1 \bmod 8)-(x2 \bmod 8) \neq 0 \qquad \text{(Equ. 3)}$$

$$cy= (y1 \bmod 8)-(y2 \bmod 8) \neq 0 \qquad \text{(Equ. 4)}$$

Since second compression is not aligned with the first, it is assumed that the original part of forged image exhibits regular blocking artifacts while the pasted one does not. These techniques mostly use linear characteristics of DCT coefficients & checks for all possible 8×8= 64 alignments in vertical & horizontal directions to maximize correlations.

Z Qu et al[17] denoted relation between input block Sm,n and output block $\hat{S}_{m,n}$ of non aligned JPEG double compression using equation (5). They expressed output block $\hat{S}_{m,n}$ as a linear mixture of four input blocks {Sm,n, Sm,n+1, Sm+1,n, Sm+1,n+1} that it overlaps.

$$\hat{S}_{m,n} = \sum_{i=0}^{1} \sum_{j=0}^{1} A_{cy,i} S_{m-i,n-j} A^{T}_{cx,j} + \hat{E}_{m,n} \qquad \text{(Equ 5)}$$

Where $\hat{E}_{m,n}$ represents quantization noise of the second JPEG compression. $\{A_{cx,0},A_{cx,1},A_{cy,0},A_{cy,1}\}$ represents set of mixing matrices. Their coefficients ware determined by the shifted distance (cx, cy) and the DCT transform matrix. De-mixing of Equation (5) supposed to be achieved when the NADJPEG image is shifted back to its original block segmentation.

In Ghost detection technique difference between given tampered image & its various compressed versions of different quality are searched for minima which appear as dark ghost. Since it is difficult for human being to scan all difference images, automation is achieved by coinciding ghost with one of the segmented object from image. As forgery is created by copying object from source image to destination image there are more chances that ghost will coincide with one of the object in image.

In section 2 we have restated Farid's ghost detection [15] & discussed difficulties in its implementation. In section 3 we have proposed Automation Algorithm & showed practical implementation in section 4. Finally we have concluded in section 5.

II. FARID'S GHOST DETECTION

Farid's ghost detection [15] is dependent on his experiment [16] in which a set of DCT JPEG coefficients c1 quantized by an amount q1 get subsequently quantized a second time by an amount q2 yielding coefficients c2, the difference between c1 and c2 is minimal when q2 = q1 and increases as the difference between q2 and q1 go on increasing (except if q2 = 1 i.e., no second quantization). Specifically, if q2 > q1 then the coefficients c2 become increasingly more sparse relative to c1, and if q2 < q1 then, even though the second quantization is less than the first, the coefficients c2 shift relative to c1. If each DCT coefficient is compared in

YCbCr channel differently, multiple minima may occur so they considered the cumulative effect of quantization on the underlying pixel values. In order to compensate between low & high frequency region present in image they considered spatially averaged and the normalized difference measure

Farid computed difference directly from the pixel values instead of computing the difference between the quantized DCT coefficients. Thus they avoided possible multiple minima at each color channel.

$$d(x,y,q) = \frac{1}{3}\sum_{i=1}^{3}[f(x,y,i) - f_q(x,y,i)]^2 \qquad \text{(Equ 6)}$$

Where f (x, y, i), i = 1, 2, 3, denotes each of three RGB color channels, and fq (·) is the result of compressing f (·) at quality q. Shown in the top left panel of the figure 2 is an image whose central 200×200 pixel region was extracted, compressed at a JPEG quality of 65/100, and re-inserted into the image whose original quality was 85. Shown in each subsequent panel is the sum of squared differences, Equation (6), between this manipulated image, and a re-saved version compressed at different JPEG qualities. Central region is clearly visible when the image is re-saved at the quality of the tampered region (65) and overall error reaches a minimum at the saved quality of 85.

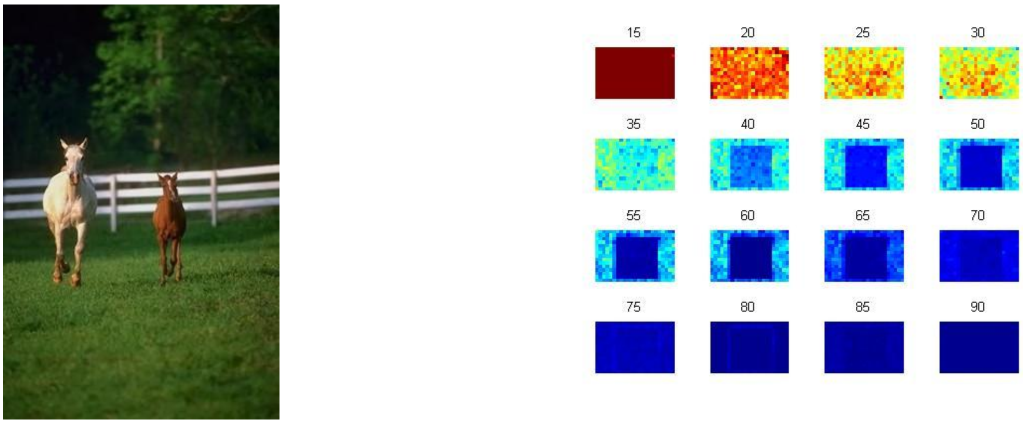


Figure 2. Difference images

In order to compensate effects of low & high frequency regions difference image is first averaged across a b × b pixel region by Equation(7) & then normalized as Equation (8) so that the averaged difference at each location (x,y) is scaled into the range[0,1].

$$\delta(x,y,q) = \frac{1}{3}\sum_{i=1}^{3}\frac{1}{b^2}\sum_{b_x=0}^{b-1}\sum_{b_y=0}^{b-1}[f(x+b_x, y+b_y, i) - f_q(x+b_x, y+b_y, i)]^2 \qquad \text{(Equ 7)}$$

$$d(x,y,q) = \frac{\delta(x,y,q) - \min_q[\delta(x,y,q)]}{\max_q[\delta(x,y,q)] - \min_q[\delta(x,y,q)]} \qquad \text{(Equ 8)}$$

III. EXPERIMENTAL EVALUATION OF GHOST DETECTION

Farid has given the result only for the forgery created from same image that too only central region was forged. Also the JPEG quality difference between central forged area & outer unforged area was minimum 20. When we tried to search for the ghost in blind forged images of CASIA Database V.2 [18] we observed following problems.

1. Theoretically Farid's [15] ghost mechanism is correct but practical implementation in same form is very difficult.
2. If quality of ghost image area (forged area) & surrounding image area is same ghost cannot be detected as everywhere in the forged image difference will come out as minimum.

3. If area of ghost region is very small as compared to complete forged image it becomes difficult to differentiate between actual ghost & other dark spot arriving because of actual low intensity values in original image..
4. If un-tampered image consists of low intensity area it may also come out as ghost since intensity difference in that area will be again very low. From above findings it is clear that we need a technique to identify whether a ghost is real ghost or it is arriving just because of low intensity area in an image. User needs to analyse multiple image at different compression quality for single forged image so it becomes very difficult to validate technique against huge forgery database.

   In practice, the amount of human interaction is extremely time consuming as a ghost can be visually hard to distinguish from noise & number of difference images can become very large

IV. PROPOSED GHOST AUTOMATION ALGORITHM

In this proposed algorithm first suspected image is segmented into different segments by using Graph based segmentation. Graph based method is based on selecting edges from a graph, where each pixel corresponds to a node in the graph, and certain neighbouring pixels are connected by undirected edges. Weights on each edge measure the dissimilarity between pixels. Technique adaptively adjusts the segmentation criterion based on the degree of variability in neighbouring regions of the image. Suspected image is recompressed at different quality levels & subtracted from original image. Subtraction is performed at each individual RGB color channel & effective average of three color channel is considered. Difference images are computed by using Farid's approach, morphologically processed & converted to black & white. If ghost appears in the resultant image it will come out as one large component whose size should not be too large to span complete image & too small to appear as noise throughout the image. So we considered only those difference images where size of component is greater than 1/8 of minimum size segment & less than twice of maximum size segment of original segmented image. All remaining difference images are discarded. This ghost will overlap with one of the segment we got during segmentation phase. If ghost size is within the range of corresponding segment size (here we have assumed it to be 5000 pixels) ghost is valid ghost & image identified as tampered. Complete Automation Algorithm is as given below.

Ghost Validation using Graph based Segmentation Algorithm (Image I)

```
1. Iseg= Graph_Seg (I) // Segment suspected Image I
2. Max_Segment_size= Size of Image segment with maximum size
3. Min_Segment_size= Size of image Segment with minimum size
4. Tampered=0
5. For q=1:Q  // Quality of JPEG Image
   a. Recompress I at JPEG quality q to get image Iq
   b. $I_b$=I-$I_q$ // Subtract recompressed image from original compresses image I.
   c. Average image Iq by moving b×b size window.
   d. Normalize the Iq between 0 to1
   e. Convert $I_b$ to black & white Image
   f. Perform Opening in Image $I_b$
   g. ghost_size= Size of largest Component present in image $I_b$
   h. (If ghostsize>=Min_Segment_size/8 && ghostsize <= Max_Segment_size × 2)
       i. seg_ghost_size=segment size of Iseg overlapping with Ib
      ii. if |seg_ghost_size – ghostsize| <=5000
          tampered=1
   i. end-if
6. end-for
```

V. EXPERIMENTAL EVALUATION

We used CASIA V.2 [18] tampered image database for evaluation which consist of different types of images such as animal, architect, art, character, nature. One of the tampered image is as shown in rightmost image of figure3 which is created after splicing two rightmost images of figure3. Results at different stages of algorithm are shown from figure 4-6. Figure 4 represents segmented image we got after applying Graph based segmentation. Figure 5 represents difference images we got after subtracting image from different recompressed images. Figure 6 represents those difference images in which ghost is coinciding with segment of image. Figure 7 shows the final image in which ghost has the sufficient size compared to corresponding matching segment showing tampered area in image.

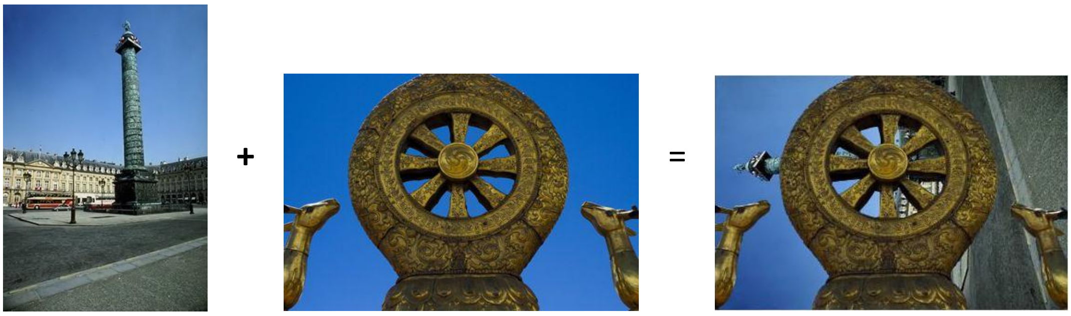
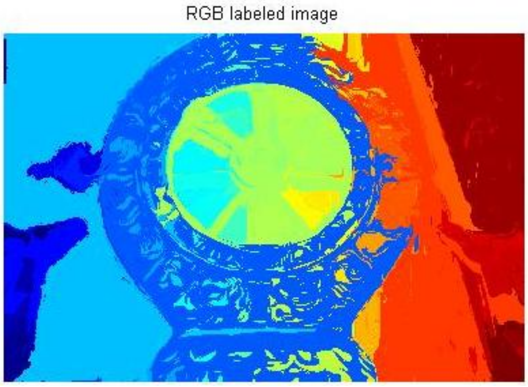


Figure 3. Creation of Tampered Image



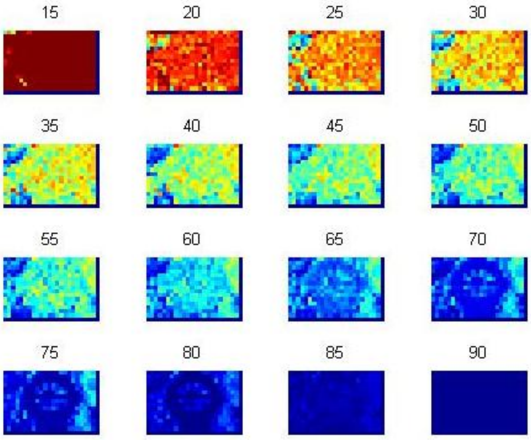
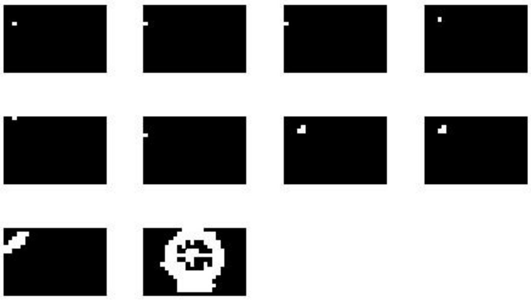


Figure 4. Segmented Image

Figure 5. Difference Image



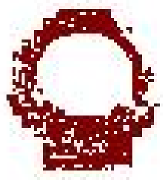


Figure 6. Difference Image Coincided with Segments of Image

Figure 7. Automatically identified spliced Area

We have tested this algorithm against those JPEG images of CASIA tampered image database which are created by splicing two JPEG images. 80% of the images in which ghost (dark region) is manually visible in difference images ware detected automatically. But overall forgery detection rate was very poor. This occurs because in most of the tampered images pasted region undergoes NADJPEG compression so it doesn't come out as dark region rather it come out as lighter region as compared to surrounding region which undergo ADJPEG compression. Also in many images ghost doesn't arrive in difference images because of which forgery detection rate decreases.

VI. CONCLUSION

Here we have proposed a technique for automated ghost detection which will avoid user to scan large number of difference images. This technique will automatically give result as true or false & also will give forged area in image. In preliminary analysis we got 19 % false acceptance rate. As ghost is considered as small dark region in difference images some of the tampered images where surrounding area comes out as dark(undergo ADJPEG compression) while pasted region comes out as lighter area(undergo NADJPEG compression) & pass in automation process undetected. This drawback can be removed if intelligent approach for selection of ghost is considered (dark or light). In future we will try to classify ghost based on size of dark & light region which will improve False Rejection Rate.

REFERENCES

[1] A.C. Popescu and H. Farid, “Exposing digital forgeries by detecting traces of re-sampling,” IEEE Transactions on Signal Processing, vol. 53, no. 2, pp. 758–767, 2005.

[2] B. Mahdian and S. Saic, “Blind authentication using periodic properties of interpolation,” IEEE Transactions on Information Forensics and Security, vol. 3, no. 3, pp. 529–538, 2008.

[3] S. Prasad and K. R. Ramakrishnan, “On resampling detection and its application to image tampering,” in Proceedings of the IEEE International Conference on Multimedia and Exposition, Toronto, Canada, 2006, pp.1325–1328.

[4] Sevinc Bayram,Husrev Sencar, Nasir Memon,” Identifying Digital Cameras Using CFA Interpolation ”, IFIP international Conference on Digital Forensics, Orlando, Florida , 2006 ,Volume 222, pp 289-299.

[5] Y. Long , Y. Huang, “Image based source camera identification using demosaicking”, IEEE International Workshop on Multimedia Signal Processing, 2006, vol. 3 ,pp 419-424

[6] Kakar, P., Sudha, N. & Ser, W., “Exposing Digital Image Forgeries by Detecting Discrepancies in Motion Blur”, IEEE Transactions on Multimedia, 2011, Vol. 13(3), pp. 443-452

[7] Q. Liu, X. Cao, C. Deng, and X. Guo, “Identifying image composites through shadow matte consistency,” IEEE Transaction on Information. Forensics Security, vol. 6, no. 3, pp. 1111–1122, Sep. 2011.

[8] Criminisi, I. Reid, and A. Zisserman, “Single view metrology,” Int. J. Computer Vision, vol. 40, no. 2, pp. 123–148, Nov. 2000.

[9] M. K. Johnson and H. Farid, “Detecting photographic composites of people,” Proc. Int. Workshop on Digital Watermarking, Guangzhou, China, 2007.

[10] W. Zhang, X. Cao, Y. Qu, Y. Hou, H. Zhao, and C. Zhang,“Detecting and extracting the photo composites using planar homography and graph cut,” IEEE Trans. Inf. Forensics Secur., vol. 5, pp. 544–555, Sep. 2010.

[11] M Johnson, H. Farid ,”Exposing digital forgeries through chromatic aberration” Proceeding of multimedia & security workshop , 2006,pp 48-55.

[12] Archana V. Mire, Dr S. B. Dhok, Dr N. J. Mistry, Dr P. D. Porey, “Catalogue of Digital Image Forgery Detection Techniques, An Overview ”, in proceeding of Third International Conference on Advances in Information Technology and Mobile Communication – AIM 2013.

[13] Babak Mahdian, Stanislav Saic, “A bibliography on blind methods for identifying image forgery”, Signal Processing: Image Communication 25 (2010) 389–399

[14] Hany Farid, ”Image Forgery Detection, A survey”, IEEE signal processing Magazine, March 2009

[15] H. Farid, “Exposing digital forgeries from JPEG ghosts”, IEEE Trans. Inf. Forensics Security 4(1), 154–160 (2009).

[16] www.cs.dartmouth.edu/farid/downloads/.../digitalimageforensics .pdf

[17] Z. Qu, W. Luo, and J. Huang, “A convolutive mixing model for shifed double JPEG compression with application to passive image authentication,” in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP ’08), pp. 1661–1664, IEEE, Las Vegas, Nev, USA, March-April 2008.

[18] Casia Tampered Image Database V.2 http://forensics.idealtest.org:8080/index_v2.html