

Design of Classifier for Detecting Image Tampering Using Gradient Based Image Reconstruction Technique

Sonal Sharma, Preeti Tuli

Abstract

Image tampering detection is a significant multimedia forensics topic which involves, assessing the authenticity or not of a digital image. Information integrity is fundamental in a trial but it is clear that the advent of digital pictures and relative ease of digital image processing makes today this authenticity uncertain. In this paper this issue is investigated and a framework for digital image forensics is presented, individuating if the tampering has taken place. Based on the assumptions that some processing must be done on the image before it is tampered, and an expected distortion after processing an image, we design a classifier that discriminates between original and tampered images. We propose a novel methodology based on gradient based image reconstruction to classify images as original or tampered. This methodology has its application in a context where the source image is available (e.g. the forensic analyst has to check a suspect dataset which contains both the source and the destination image).

Index Terms — Gradient, Poisson equation, Region of interest (ROI), Digital image forensics, Authenticity verification, Image reconstruction from gradients.

1 INTRODUCTION

In today's digital age, the creation and manipulation of digital images is made simple by digital processing tools that are easily and widely available. As a consequence, we can no longer take the authenticity of images for granted especially when it comes to legal photographic evidence. *Image forensics*, in this context, is concerned with determining the source and potential authenticity of an image. Although digital watermarks have been proposed as a tool to provide authenticity to images, it is a fact that the overwhelming majority of images that are captured today do not contain a digital watermark. And this situation is likely to continue for the foreseeable future. Hence in the absence of widespread adoption of digital watermarks, there is a strong need for developing techniques that can help us make statements about the origin, veracity and authenticity of digital images.

In this paper we focus on the problem of reliably discriminating between "tampered" images (images which are altered in order to deceive people) from untampered original ones. The basic idea behind our approach is that a tampered image (or the least parts of it) would have undergone some image processing. Hence, we design a classifier that can distinguish between images that have and have not been processed. We apply it to a suspicious image of a target image and classify the suspicious image as tampered or untampered. The rest of this paper is organized as follows: In Section 2 we present a method to verify the authenticity of images that is used in the classifier we design for image forensics, i.e. we formulate the problem and present solution methodology. Statistical performance results are given in Section 3, with conclusions drawn in section 4.

2 PROBLEM FORMULATION AND SOLUTION METHODOLOGY

The problem of fraud detection has been faced by proposing different approaches each of these based on the same concept: a forgery introduces a correlation between the original image and the tampered one. Several methods search for this dependence by analyzing the image and then applying a feature extraction process. In [1] the direct approach proposed by Fridrich et al. comprises of performing an exhaustive search by comparing the image to every cyclic – shifted versions of it, which requires (MN^2) steps for an image sized M by N. This computationally expensive search does not work where the copied region has undergone modifications. In [2] A.N. Myma et al. presented an approach of first applying wavelet transform to the input image to yield a reduced dimension representation, then exhaustive search is performed to identify similar blocks in the image by mapping them to log polar co-ordinates and using phase correlation as the similarity criterion. But the performance relies on the location of copy- move regions. In [3] Weihai Li et al. utilized the mismatch of information of block artifact grid as clue of copy paste forgery. A DCT grid is the horizontal lines and the vertical lines that partition an image into blocks and a block artifact grid (BAG) is the grid embedded in an image where block artifact appears. The DCT grid and BAG match together in untampered images. But if the copied area is from the other different image it cannot be detected by the method, also the complexity of algorithm is high. In [4] Bayram et al. proposed Fourier – Mellin transform (FMT). But the algorithm works for the case of only slight rotation. In [5] Xu Bo et al. proposed a method in which Speed up Robust features (SURF) key points are extracted and their descriptors are matched within each other with a threshold value. This method fails to automatically locate the tampered

region and its boundary.

None of these approaches [1, 2, 3, 4, and 5] conducts authentication verification using gradient maps in the image reconstruction. The approach presented in this paper verifies the authentication in two phases *modeling phase* and *simulation phase*. In *modeling phase* the image is reconstructed from the image gradients by solving a poisson equation and in the *simulation phase* absolute difference method and histogram matching criterion between the original and test image is used. The solution methodology is discussed in the subsequent paragraphs.

2.1 Image Reconstruction

In the year 1993 Luc Vincent [6] carried out the work in morphological grayscale reconstruction. In 2004 Di Zang and G. Sommer [7] carried out phase based image reconstruction in the monogenic scale space. In 2005 S. Leng et al. [8] presented fan-beam image reconstruction algorithm to reconstruct an image via filtering a back projection image of differentiated projection data. In 2008 A. L. Kesidis and N. Papamarkos [9] presented a new method for the exact image reconstruction from projections. The original image is projected into several view angles and the projection samples are stored in an accumulator array. In 2011 P. Weinzaepfel et al [10] proposed another novel approach which consists using an off-the-shelf image database to find patches visually similar to each region of interest of the unknown input image.

The approach presented in this paper is gradient based image reconstruction by solving poisson equation. The details of the method are described as under:

2.1.1 Gradient Based Image Reconstruction:

As already stated in our previous work [11] image reconstruction from gradient fields is a very active research area. The gradient-based image processing techniques and the poisson equation solving techniques have been addressed in several related areas such as high dynamic range compression [12], Poisson image editing [13], image fusion for context enhancement [14], interactive photomontage [15], Poisson image matting [16] and photography artifacts removal [17]. A new criterion is developed, where the image is reconstructed from its gradients by solving a poisson equation and hence used for authenticity verification [11].

In 2D, a modified gradient vector field,

$$G' = [G'_x, G'_y]$$

(1) may not be integrable.

Let I' denote the image reconstructed from G' , we employ one of the direct methods recently proposed in [12] to minimize,

$$\|\nabla I' - G'\|$$

(2) so that,

$$G' \approx \nabla I'$$

(3) By introducing a Laplacian and a divergence operator, I' can be obtained by solving the Poisson differential equation [18, 19]
 $\nabla^2 I' = \text{div}([G'_x, G'_y])$

Since both the Laplacian and *div* are linear operators, approximating those using standard finite differences yields a large system of linear equations. The full multigrid method [20] is used to solve the Laplacian equation with Gaussian-Seidel smoothing iterations. For solving the poisson equation more efficiently, an alternative is to use a rapid poisson solver, which uses a sine transform based on the method [18] to invert the laplacian operator. Therefore, the rapid poisson solver is employed in our implementation. The image is zero-padded on all sides to reconstruct the image.

2.1.2 Poisson Solvers:

A Poisson solver produces the image whose gradients are closest to the input manipulated gradient domain image in a least squares sense, thereby doing a kind of inverse gradient transform. Note that if the input were a gradient domain image whose gradients had not been manipulated, the inverse gradient transformation would have an exact solution, and the poisson equation would give a perfect reconstruction of the image. Both the FFT-based solver and the poisson solver using zero Dirichlet boundary condition work successfully in obtaining an inverse gradient transformation in the sense that they give a perfect reconstruction of the image when the input gradient domain image is not manipulated. This section details the poisson solver which has been used in the present research work.

In this section, we describe the standard gradient integration problem and its poisson solution and then expand this result to include a data function term. The problem of computing a function $f(x,y)$ whose gradient $\nabla f(x,y)$ is as close as possible to a given gradient field $g(x,y)$ is commonly solved by minimizing the following objective:

$$\iint \|\nabla f - g\|^2 dx dy.$$

Note that g is a vector-valued function that is generally not a gradient derived from another function. (If g were derived from another function, then the optimal f would be that other function, up to an unknown constant offset.)

It is well-known that, by applying the Euler-Lagrange equation, the optimal f satisfies the following Poisson equation:

$$\nabla^2 f = \nabla \cdot g,$$

(6) which can be expanded as $f_{xx} + f_{yy} = g_x^x + g_y^y$, where $g = (g^x, g^y)$. Subscripts in x and y correspond to partial derivatives with respect to those variables. We have superscripted g^x and g^y to denote the elements of g rather than subscript them, which would incorrectly suggest they are partial derivatives of the same function. We now expand the objective beyond the standard formulation. In particular, we additionally require $f(x, y)$ to be as close as possible to some data function $u(x, y)$. The objective function to minimize now becomes:

$$\iint \lambda_d (f - u)^2 + \|\nabla f - g\|^2 dx dy,$$

(7) Where, λ_d is a constant that controls the trade-off between the fidelity of f to the data function versus the input gradient field. To solve for the function f that minimizes this integral, we first isolate the integrand:

$$L = \lambda_d (f - u)^2 + \|\nabla f - g\|^2 = \lambda_d (f - u)^2 + (f_x - g^x)^2 + (f_y - g^y)^2$$

(8) The function f that minimizes this integral satisfies the Euler-Lagrange equation:

$$(dL / df) - (d/dx) \cdot (dL/df_x) - (d/dy) \cdot (dL/df_y) = 0$$

(9) Substituting and differentiating, we then have:

$$2 \lambda_d (f - u) - 2(f_{xx} - g^x) - 2(f_{yy} - g^y) = 0$$

(10) Rearranging gives us:

$$\lambda_d f - (f_{xx} + f_{yy}) = \lambda_d u - (g_{xx} + g_{yy})$$

(11) or equivalently:

$$\lambda_d f - \lambda^2 f = \lambda_d u - \lambda \cdot g$$

(12)

The left-hand side of this equation is poisson equation, typically studied in three dimensions in physics. Our analysis will be in 2D. As expected, setting $\lambda_d = 0$ nullifies the data term and gives us the Poisson equation.

2.1.3 Discrete Sine Transform

In this section we analyze the 2D Poisson equation in the sine domain. As with fast Poisson solvers, we can solve the screened poisson equation (Equation 8) by taking its sine transform.

The **discrete sine transform** (DST) is a Fourier-related transform similar to the discrete Fourier transform (DFT), but using a purely real matrix. It is equivalent to the imaginary parts of a DFT of roughly twice the length, operating on real data with odd symmetry (since the Fourier transform of a real and odd function is imaginary and odd), where in some variants the input and/or output data are shifted by half a sample.

Formally, the discrete sine transform is a linear, invertible function $F : \mathbb{R}^N \rightarrow \mathbb{R}^N$ (where \mathbb{R} denotes the set of real numbers), or equivalently an $N \times N$ square matrix. There are several variants of the DST with slightly modified definitions. The N real numbers x_0, \dots, x_{N-1} are transformed into the N real numbers X_0, \dots, X_{N-1} according to the formula:

$$X_k = \sum_{n=0}^{N-1} x_n \sin \left[\frac{\pi}{N+1} (n+1)(k+1) \right] \quad k = 0, \dots, N-1 \tag{13}$$

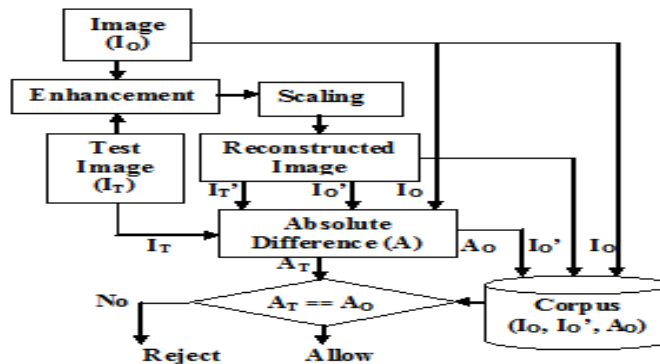


Figure 1: Schematic diagram for Modeling and Simulation Phase

2.1.4 Inverse Discrete Sine Transform

The inverse of DST is DST multiplied by $2/(N+1)$. Like for the DFT, the normalization factor in front of these transform definitions is merely a convention and differs between treatments.

2.2 Absolute Difference

In the present work our approach is to find the absolute difference between the original and the reconstructed image. Subtraction gives the difference between the two images, but the result may have a negative sign and can be lost. The function that finds how different the two images are- regardless of the arithmetic sign- is the absolute difference:

$$N(x, y) = |O_1(x, y) - O_2(x, y)|,$$

(14) where, $O_1(x, y)$ and $O_2(x, y)$ are pixels in the original images, $|x|$ is the absolute difference operator, and $N(x, y)$ is the resultant new pixel. The absolute difference operator returns $+x$ whether the argument is $-x$ or $+x$.

2.3 Histogram Normalization

Histogram is a graphical representation of the intensity distribution of an image. It quantifies the number of pixels for each intensity value considered. Histogram normalization is a method that improves the contrast in an image, in order to stretch out the intensity range. Equalization implies mapping one distribution (the given histogram) to another distribution (a wider and more uniform distribution of intensity values) so that the intensity values are spread over the whole range.

To accomplish the equalization effect, the remapping should be the cumulative distribution function (CDF)

For the histogram $H(i)$, its cumulative distribution $H'(i)$ is:

$$H'(i) = \sum_{j=0}^i H(j), \text{ where } 0 \leq j < i$$

(15) To use this as a remapping function, we have to normalize $H'(i)$ such that the maximum value is 255 (or the maximum value for the intensity of the image). Finally, we use a simple remapping procedure to obtain the intensity values of the equalized image:

$$\text{equalized}(x, y) = H'(\text{src}(x,y)) \quad (16)$$

In our work, first we perform the histogram normalization and then the histogram equalization criteria is used where the normalized histogram values of the original and test image are utilized for matching the two images.

2.4 Algorithm Used:

Algorithm 1: Modeling and Simulation of original and reconstructed image

Modeling phase

Step 1: Read an image (I_O).

Step 2: Convert into grayscale image, say R.

(Enhancement stage)

Step 3: Perform Scaling on the image.

Step 4: Enhance the image using median filtering and convolution theorem (I_O).

Step 5: Reconstruct the image using proposed methodology (I_O').

Step 6: Find the absolute difference between original and reconstructed image (A_O).

Step 7: Store the original image, reconstructed image and absolute difference (I_O, I_O', A_O)

Simulation phase

Step 8: Input a test image (I_T)

Step 9: Reconstruct I_T to obtain I_T' and find the absolute difference (A_T) between I_T and I_T'

Step 10: Compare A_T and A_O to find a match and hence allow or reject the subject accordingly.

2.5 Modeling and Simulating

As shown in Fig. 1, in the modeling phase, let I_O be the original image of a subject which has to be modeled for the formation of knowledge based corpus. After enhancing and proper scaling of the original image I_O , the image is poisson reconstructed from its gradients as:

$$I_O' = \text{Poisson_reconstruction}(I_O)$$

(17) Now the absolute difference between the original and reconstructed image is calculated as:

$$A_O = \text{Absolute_difference}(I_O, I_O')$$

(18) Now store the triplet (I_O, I_O', A_O) in the corpus so as to form the knowledge based model (corpus). The equations (17) and (18) can be repeatedly used to register n number of subjects, and store their details for authentication verification.

In the simulation phase, when the tampered or forged image will be presented to the security system for authentication, the system will reconstruct the test image (I_T) as:

$$I_T' = \text{Poisson_reconstruction}(I_T)$$

(19) And, then the absolute difference between the original test image (I_T) and reconstructed test image (I_T') is calculated as:

$$A_T = \text{Absolute_difference}(I_T, I_T')$$

(20) Now, the resultant A_T is compared with A_O (the absolute difference stored in corpus of the original and reconstructed original image in modeling phase)

If ($A_T == A_O$)
 “Authenticity Verified as TRUE!”
 Else
 “Authenticity Verified as FALSE!”

Hence, the result will reject the subject due to a mismatch and the images obtained by forgery or tampering for authenticity verification will be classified as fake or invalid and any hidden data (for destroying the security system or secret communication) will be clearly identified.

3 RESULTS AND DISCUSSION

The solution methodology for the above stated problem is implemented using proposed algorithm and the experimental outcomes are shown below:

3.1 Results for modeling phase (Original Image).



Figure 2.1: Original Image (I_O)

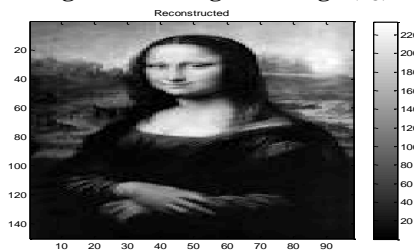


Figure 2.2: Reconstructed Image (I_O')

Solving Poisson Equation Using DStime for Poisson Reconstruction = 0.703055 secs
 (Image Size: 98x150)

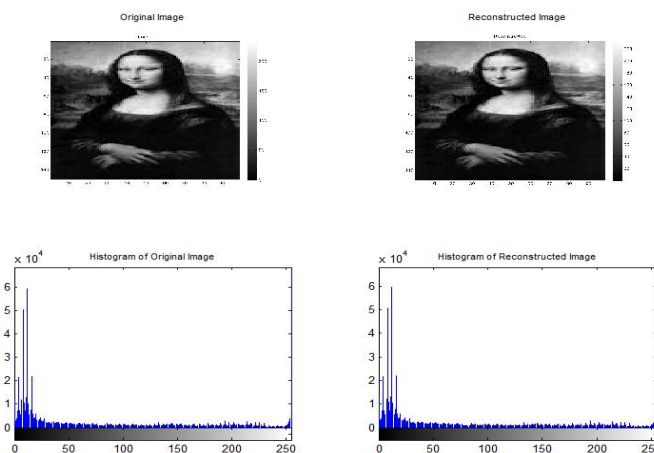
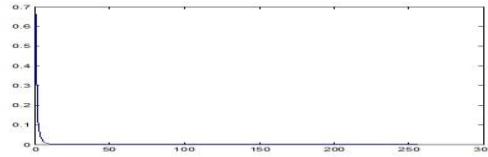


Figure 2.3: Histogram of original image (I_O) and Reconstructed Image (I_O')



Figure 2.4: Histogram of absolute difference of original image (I_O) and reconstructed original image (I_O') (left), Plot of absolute difference of original image (I_O) and reconstructed original image (I_O') (right)



As shown in Fig. 2.1 to Fig. 2.5, the registration of the images for authenticity verification has been done by using the steps of the modeling phase mentioned in Algorithm 1. The original image I_O (Fig. 2.1) is converted to grayscale image, then the image is scaled and enhanced and then the scaled image is poisson reconstructed and the resultant image (I_O') is shown in Fig. 2.2. The histogram of the original and reconstructed image is shown in subplots in Fig. 2.3. For authenticity verification the absolute difference (A_O) of the original image (I_O) and reconstructed image (I_O') is calculated. Now the histogram of A_O is obtained and the result is shown in Fig. 2.4 (left). The plot of the histogram so obtained is shown in Fig. 2.4 (right), and the plot of normalized histogram is shown in the Fig. 2.5. This plot of the normalized histogram will be compared with that of the test image during simulation. Now the corpus contains the triplet (I_O, I_O', A_O) for the registered subject's original image.

The above can be repeated for the registration of n number of subject's.

3.2 Results for simulation phase [Test (tampered) Image]



Figure 3.1: Test Image (I_T)

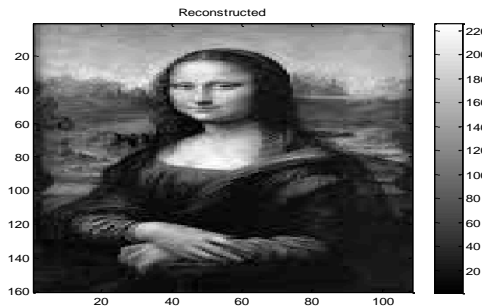


Figure 3.2: Reconstructed Test Image (I_T')
Solving Poisson Equation Using DST
Time for Poisson Reconstruction = 0.060161 secs

Image Size: (184×274)

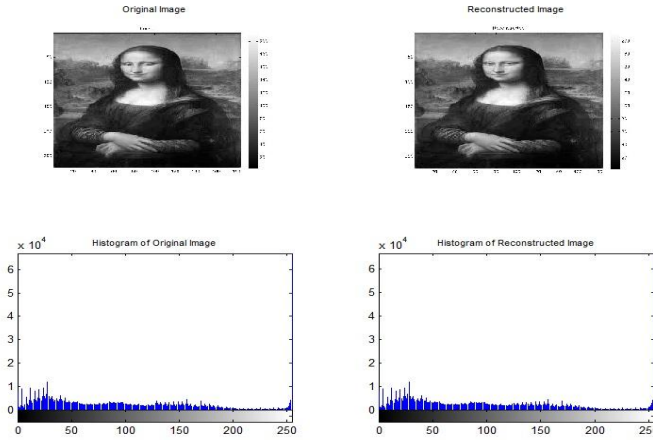


Figure 3.3: Histogram of original Test image (I_T) and Reconstructed Test Image (I_T')

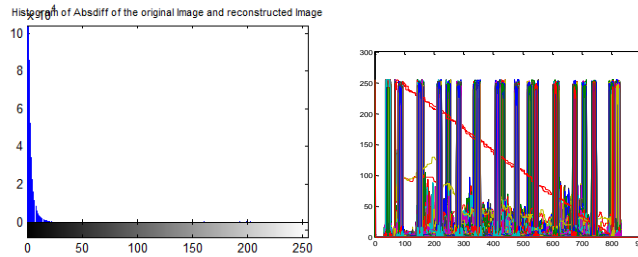


Figure 3.4 : Histogram of absolute difference of test image (I_T) and reconstructed test image (I_T') (left), Plot of absolute difference of test image (I_T) and reconstructed test image (I_T') (right)

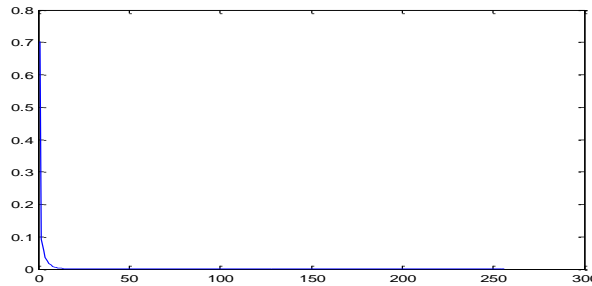


Figure 3.5: Plot of the Normalized Histogram of the absolute difference of test image (I_T) and reconstructed test image (I_T')

The test image (tampered) is passed through the steps of simulation phase mentioned in Algorithm 1 and the results are shown in Fig. 3.1 to Fig. 3.5. The authenticity verification of the test image (I_T) has been done by using the steps of the simulation phase mentioned in Algorithm 1. The test image I_T (Fig. 3.1), is converted to grayscale image, then the image is scaled and enhanced. Now the scaled image is poisson reconstructed and the resultant image (I_T') is shown in Fig. 3.2. The histogram of the test and reconstructed test image is shown in subplots in Fig. 3.3. For authenticity verification the absolute difference (A_T) of the test image (I_T) and reconstructed test image (I_T') is calculated and then the histogram of A_T is obtained and the result is shown in Fig. 3.4 (left) and the plot of the histogram so obtained is shown in Fig. 3.4 (right). Finally, the plot of normalized histogram is shown in the Fig. 3.5 which will be compared with that stored in corpus.

4 CONCLUSION

The normalized histogram of absolute difference of the test image and reconstructed test image (Fig. 3.5) is compared with the normalized histogram of absolute difference of the original image and reconstructed original image (Fig. 2.5), and the so obtained result is inequality, since, the value of the difference is not zero and comes to be- 0.0049units.

And, hence the image is classified as tampered and finally rejected. If the image was not tampered then the so obtained difference (between the the normalized histogram of absolute difference of the test image and reconstructed test image (Fig. 3.5) and the normalized histogram of absolute difference of the original image and reconstructed original image (Fig. 2.5) would be- 0.0000units.

In this manner the authenticity of the individual's can be verified and the test images can be classified as tampered (or forged) or original, and hence the tampering can be detected. We also observed that the time required to reconstruct the original image is- 0.703055 secs,

But, the time required to reconstruct the tampered image is- 0.060161 secs.

Hence, we also conclude that the tampering can be detected by the time our poisson solver takes to reconstruct the original image and test image.

REFERENCES

- [1] J. Fridrich, D. Soukal, and J. Lukas, "Detection of copy-move forgery in digital images," Proc. Digital Forensic Research Workshop, Cleveland, OH, August 2003.
- [2] A.N.Myrna, M.G.Venkateshmurthy, "Detection of Region Duplication Forgery in Digital Images Using Wavelets and Log-Polar Mapping," Conference on Computational Intelligence and Multimedia Applications, Dec. 2007, Vol.3, pp. 371-377.
- [3] Weihai Li, Nenghai Yu and Yuan Yuan, "Doctored jpeg image detection," IEEE ICMS 2008, pp 253-256.
- [4] S. Bayram, H. T. Sencar, and N. Memon, "An Efficient and Robust Method for Detecting Copy-move Forgery," Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, Taipei, Taiwan, pp. 1053-1056, 2009.
- [5] Xu Bo, Wang Junwen, Liu Guangjie and Dai Yuewei, "Image Copy-move Forgery Detection Based on SURF," International Conference on Multimedia Information Networking and Security, IEEE pp 889-892.
- [6] Luc Vincent, "Morphological Grayscale Reconstruction in Image Analysis: Applications and Efficient Algorithms," IEEE Transactions on Image Processing, vol. 2, no. 2, 1993.
- [7] Di Zang and G. Sommer, "Phase Based Image Reconstruction in the Monogenic Scale Space," DAGM-Symposium, 2004.
- [8] S. Leng, T. Zhuang, B. Nett and Guang-Hong Chen, "Exact fan-beam image reconstruction algorithm for truncated projection data acquired from an asymmetric half-size detector," Phys. Med. Biol. 50 (2005) 1805-1820.
- [9] A. L. Kesidis, N. Papamarkos, "Exact image reconstruction from a limited number of projections," J. Vis. Commun. Image R. 19 (2008) 285-298.
- [10] P. Weinzaepfel, H. Jegou, P. Perez, "Reconstructing an image from its local descriptors," Computer Vision and Pattern Recognition (2011).
- [11] S. Sharma, P. Tuli, "Fraud and Tamper Detection in Authenticity Verification through Gradient Based Image Reconstruction Technique for Security Systems," IOSR Journal of Computer Engineering (IOSRJCE) ISSN: 2278-0661 Volume 3, Issue 4 (July-Aug. 2012), PP 01-06
- [12] R. Fatta, D. Lischinski, M. Werman, "Gradient domain high dynamic range compression" ACM Transactions on Graphics 2002;21(3):249-256.
- [13] P. P´erez, M. Gangnet, A. Blake, "Poisson image editing" ACM Transactions on Graphics 2003;22(3):313-318.
- [14] R. Raskar, A. Ilie, J. Yu, "Image fusion for context enhancement and video surrealism", In: Proceedings of Non-Photorealistic Animation and Rendering '04, France, 2004. p. 85-95.
- [15] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, M. Cohen M, "Interactive digital photomontage. ACM Transactions on Graphics" 2004;23(3):294-302.
- [16] J. Sun, J. Jia, CK. Tang, HY Shum, "Poisson matting. ACM Transactions on Graphics" 2004;23(3):315-321.
- [17] A. Agrawal, R. Raskar, SK. Nayar, Y. Li, "Removing flash artifacts using gradient analysis" ACM Transactions on Graphics 2005;24(3):828-835.
- [18] R. Raskar, K. Tan, R. Feris, J. Yu, M. Turk "Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging" ACM Transactions on Graphics 2004;23(3):679-688.
- [19] J. Shen, X. Jin, C. Zhou, Charlie C. L. Wang, "Gradient based image completion by solving the Poisson equation," PCM'05 Proceedings of the 6th Pacific-Rim conference on Advances in Multimedia Information Processing – Volume Part I 257-268
- [20] W. Press, S. Teukolsky, W. Vetterling, B. Flannery "Numerical Recipes in C: The Art of Scientific Computing" Cambridge University Press; 1992.