International Journal of Computing and Corporate Research
ISSN (Online) : 2249-054X
Volume 4 Issue 5 September 2014
International Manuscript ID : 2249054XV4I5092014-02

EFFECTIVE IMPLEMENTATION OF WATERMARKING ON DIGITAL IMAGES

Vijayaraghavan
Professor
Dept. of Computer Science and Engineering
Shridevi Institute of Engineering and technology India

ABSTRACT

Digital image processing is the use of computer algorithms to perform image processing on digital images. As a subcategory or field of digital signal processing, digital image processing has many advantages over analog image processing [1]. It allows a much wider range of algorithms to be applied to the input data and can avoid problems such as the build-up of noise and signal distortion during processing. Since images are defined over two dimensions (perhaps more) digital image processing may be modeled in the form of multidimensional systems.

Keywords – Digital Watermarking, SIFT, Image Forensics

Introduction

Digital watermarking is also to be contrasted with public-key encryption, which also transform original files into another form. It is one of the techniques in image processing [1] to encrypt digital documents so that they become un-viewable without the decryption key. Unlike encryption, however, digital watermarking leaves the original image (or file) basically intact and recognizable. In addition, digital watermarks, as signatures, may not be validated without special software [2] [3]. Further, decrypted documents are free of any residual effects of encryption, whereas digital watermarks are designed to be persistent in viewing, printing, or subsequent re-transmission or dissemination.

Elements of a Watermarking System

A watermarking system can be viewed as a communication system consisting of three main elements: an embedder, a communication channel and a detector. Watermark information is embedded into the signal itself, instead of being placed in the header of a file or using encryption like in other security techniques, in such a way that it is extractable by the detector

[4]. Instead of directly embedding it into the host signal, the watermark Wo can be pre-coded to optimize the embedding process, i.e. to increase robustness against possible signal processing operations or imperceptibility of the watermark. This is done by an information coder which may require the original signal so [5].

The outcome of the information coding component is denoted by symbol W that, together with the original signal So and possibly a secret key K, are taken as input of the embedder. The secret key K is intended to differentiate between authorized users and unauthorized users at the detector in the absence of Kg [6]. The embedder takes in W, So and K, so as to hide W within So in a most imperceptible way with the help of K, and produce the watermarked signal Sw. Afterwards, Sw enters into the communication channel where a series of unknown signal processing operations and attacks may take place. The outcome of the communication channel is denoted by the symbol S´w. At the receiving end, the detector works in an inversely similar way as the embedder, and it may require the secret key Kg, K, and the original signal So. Then the detector reads S´w and decides if the received signal has the legal watermark [7][8].

Watermarking techniques can be divided into four categories according to the type of document to be watermarked as follows: [1]

- Text Watermarking
- Image Watermarking
- Audio Watermarking
- Video Watermarking

In other way, the digital watermarks can be divided into three different types as follows: [1,2]

- Visible watermark
- Invisible-Robust watermark
- Invisible-Fragile watermark

Visible watermark is a secondary translucent overlaid into the primary image. The watermark appears visible to a casual viewer on a careful inspection. The invisible-robust watermark is embedded in such a way that alternations made to the pixel value are perceptually not noticed and it can be recovered only with appropriate decoding mechanism. The invisible-fragile watermark is embedded in such a way that any manipulation or modification of the image would alter or destroy the watermark.

Also, the digital watermarks can be divided into two different types according to the necessary data for extraction:

- Informed (or private Watermarking): in which the original unwatermarked cover is required to perform the extraction process.
- Blind (or public Watermarking): in which the original unwatermarked cover is not required to perform the extraction process.

International Journal of Computing and Corporate Research
ISSN (Online) : 2249-054X
Volume 4 Issue 5 September 2014
International Manuscript ID : 2249054XV4I5092014-02

Watermarking finds its application in the following areas:

1. Broadcast Monitoring: Watermarking exists within the content itself rather than exploiting a particular segment of the broadcast signal.
2. Owner Identification : Because watermarks can be made both imperceptible and inseparable from the work that contains them, they are likely to be superior to text for owner identification. If users of the work are supplied with watermark detectors, they should be able to identify the owner of a watermarked work, even after the work has been modified in ways that would remove a textual copywright notice.
3. Transaction Tracking : The watermark records one or more transactions that have taken place in the history of the copy of a work in which it is embedded. The owner of the work would place a different watermark in each copy. If the work were subsequently misused the owner would find out who is responsible.
4. Content Authentication: Watermark can yield localized authentication and also examines whether lossy compression has been applied to the work.
5. Copy Control : Watermarks are embedded in the content itself, they are present in every representation of the content. If every recording device were fitted with a watermark detector, the devices could be made to prohibit recording whenever a never-copy watermark is detected at its input.
6. Device Control : Copy control falls into a broader category of applications, which we refer to device control.A unique identifier is embedded into printed and distributed images such as magazine advertisements , tickets etc. After the the image is recaptured by a digital camera , the watermark is read by the software on PC and the identifier is used to direct a web browser to an associated web site.

Image watermarking is the process of inserting hidden information in an image by introducing modifications to its pixels with the expectation of minimum perceptual disturbance. Watermarking is robust but still these could be the possible sources of attacks:

- Enhancement: sharpening, contrast, color correction
- Additive and multiplicative noise: Gaussian, uniform, speckle
- Linear filtering: lowpass, highpass, bandpass

- Nonlinear filtering: median filters, rank filters, morphological filter

Feature Points

In image processing the concept of feature is used to denote a piece of information which is relevant for solving the computational task related to a certain application. More specifically, features can refer to

- The result of a general neighborhood operation (feature extractor or feature detector) applied to the image,
- Specific structures in the image itself, ranging from simple structures such as points or edges to more complex structures such as objects.

Feature extraction

In pattern recognition and in image processing, feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (e.g. the same measurement in both feet and meters) then the input data will be transformed into a reduced representation set of features (also named features vector).

Corner detection

Corner detection is an approach used within image processing systems to extract certain kinds of features and infer the contents of an image. Corner detection is frequently used in motion detection, image registration, video tracking, image mosaicing, panorama stitching, 3D modelling and object recognition.

Structure of the watermark

We now give a high-level overview of our basic watermarking scheme; many variations are possible. In its most basic implementation, a watermark consists of a sequence of real numbers $X = x_1; : : : ; x_n$. In practice, we create a watermark where each value xi is chosen independently according to N(0; 1) (where $N(\mu; \sigma^2)$ denotes a normal distribution with mean μ and variance σ^2). Assume that numbers are represented by a reasonable but finite precision and ignore these insignificant round off errors. This procedure exploits the fact that each component of the watermark is chosen from a normal distribution. Alternative distributions are possible, including choosing xi uniformly from {1;-1}, {0; 1} or [0; 1].

Description of the watermarking procedure

Extract from each document D a sequence of values $V = v_1 \ldots v_n$, into which we insert a watermark X = x1; : : : ; xn to obtain an adjusted sequence of values $V` = v`_1 \ldots v`_n$. V` is then inserted back into the document in place of V to obtain a watermarked document D`. One or more

attackers may then alter D`, producing a new document D*. Given D and D*, a possibly corrupted watermark $X^{\perp}$ is extracted and is compared to X for statistical significance. Extract X* by first extracting a set of values V* = $v^*_{1.....}$ v^*_n from D* (using information about D) and then generating X* from V * and V.

While inserting X into V to obtain V` we specify a scaling parameter which determines the extent to which X alters V . Three natural formulae for computing V` are:

$v` = v_i + ax_i$ *(i)*

$v` = v_i (1 + ax_i)$ *(ii)*

$v`_I = v_i(e^{axi})$ *(iii)*

Equation 1 is always invertible, and Equations 2 and 3 are invertible if v_i 6 ≠ 0, which holds in all of our experiments. Given *V** we can therefore compute the inverse function to derive $X^{\perp}$ from *V** and *V*.

Equation 1 may not be appropriate when the v_i values vary widely. If v_i = 106 then adding 100 may be insufficient for establishing a mark, but if v_i = 10 adding 100 will distort this value unacceptably. Insertion based on Equations 2 or 3 are more robust against such differences in scale. Note that Equations 2 and 3 give similar results when xi is small. Also, when v_i is positive then Equation 3 is equivalent to $\lg(v`_i) = \lg(v_i) + x_i$, and may be viewed as an application of Equation 1 to the case where the logarithms of the original values are used.

Research Methodology

The feature point detectors used for the work to be carried out are:

- AFF (Harris Affine detector)
- HL (Hessian Laplace detector)

The two techiques used for the corner detection of feature points are:

- Harris corner detection method
- Affine Adapted corner detection method

The simulations on which the work has been carried out are:

- The time needed to detect the feature points,
- The maximum number of feature points detected.

Implementation and Results

The results are carried out on the basis of the above said techniques in MATLAB. Few gray scale images are used of which the feature points have been detected.

Harris Corner Detection method

International Journal of Computing and Corporate Research
ISSN (Online) : 2249-054X
Volume 4 Issue 5 September 2014
International Manuscript ID : 2249054XV4I5092014-02

In this, the AFF method is used to detect the feature points of the image. Harris will detect the corner feature points. The time taken by it and the number of feature points detected by this method for various images will be shown in the table below. Then the visible watermark is embedded on the detected feature points of the image. Few results of this method are:

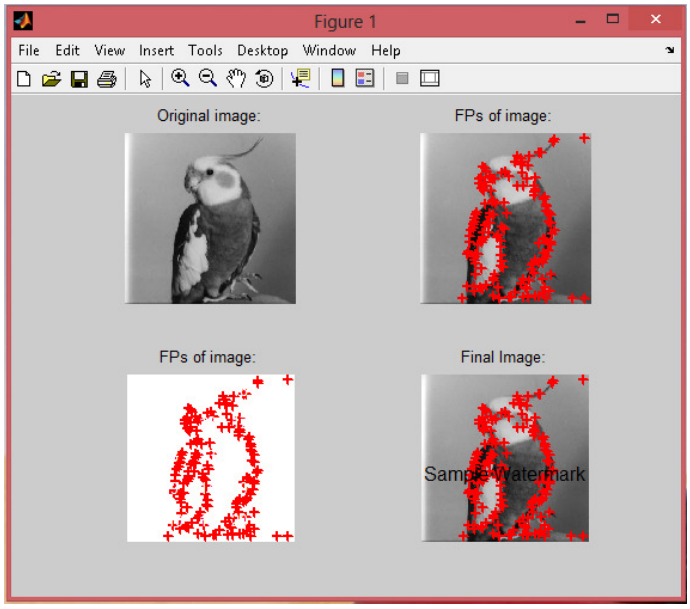
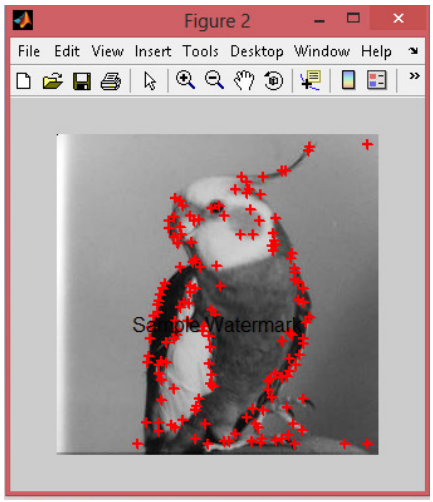


Fig. 1 Feature points detection and watermarking applied



International Journal of Computing and Corporate Research
ISSN (Online) : 2249-054X
Volume 4 Issue 5 September 2014
International Manuscript ID : 2249054XV4I5092014-02

Fig. 2 Result of feature point detected and watermark applied

In fig. 1, the image of bird has been taken. The algorithm for Harris corner detection and Harris affine detector has been applied on it. Firstly, the corner feature points have been detected and then watermarking is applied on it. In fig. 2, the final result of the feature point detected and watermarking applied on the feature points of the image has been shown. Similarly, fig. 3 and fig. 4 of Lena have been used for the same. In both the cases the number of feature points detected have been noted down and the time taken to detect the feature points and apply watermark on it has been noted. Similarly, few attempts have been carried out for the same simulation or scenario, and then the comparison report has been carried out with the second method used.

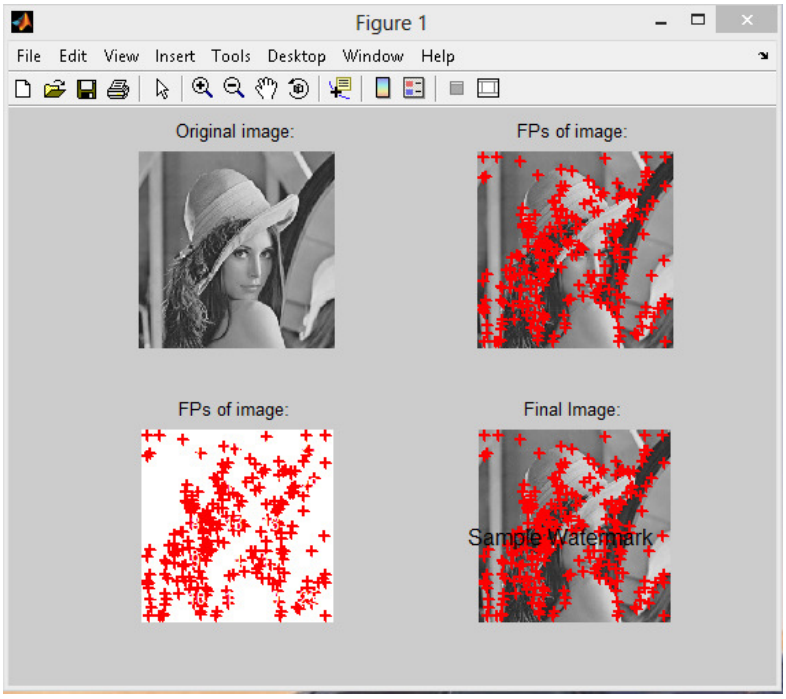


Fig. 3 Feature point and detection and watermarking is applied on original image

International Journal of Computing and Corporate Research
ISSN (Online) : 2249-054X
Volume 4 Issue 5 September 2014
International Manuscript ID : 2249054XV4I5092014-02

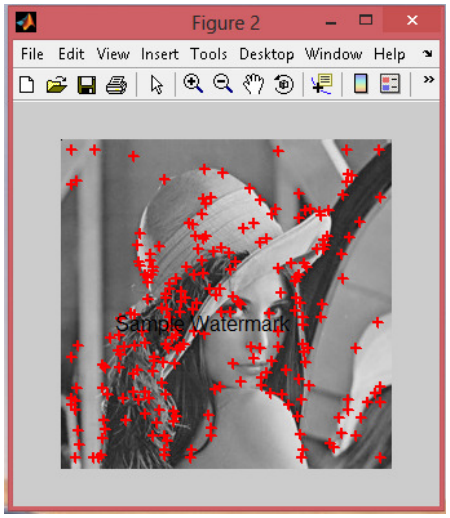


Fig 4 Result of feature point detected with watermark on it

Affine Adapted corner detection method

In this, the HL (Hessian Laplace) method is used to detect the feature points of the image. This will detect the corner feature points. The time taken by it and the number of feature points detected by this method for various images will be shown in the table below. Then the visible watermark is embedded on the detected feature points of the image. Few results of this method are:

International Journal of Computing and Corporate Research
ISSN (Online) : 2249-054X
Volume 4 Issue 5 September 2014
International Manuscript ID : 2249054XV4I5092014-02

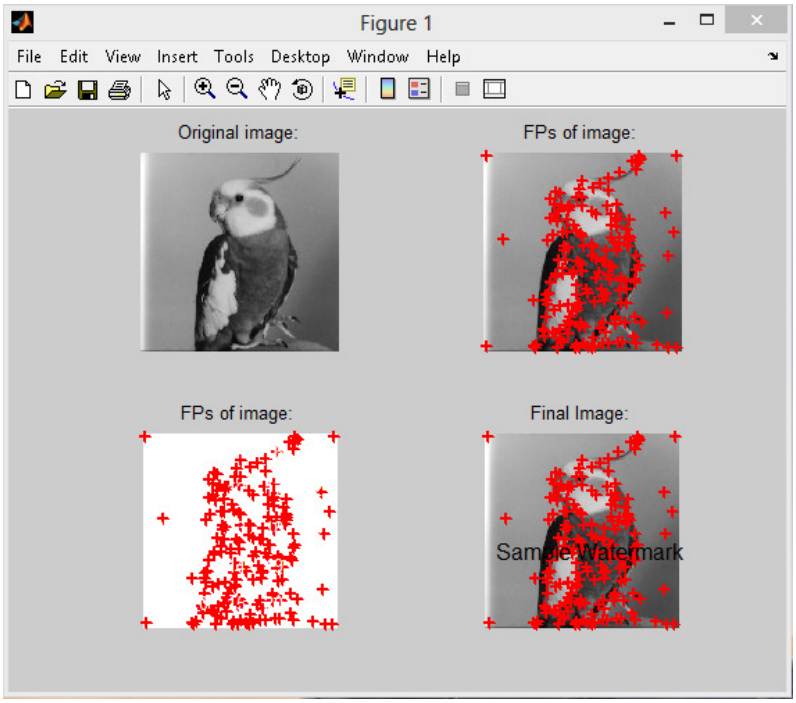


Fig. 5 The feature points are detected and watermarking is applied on the original image

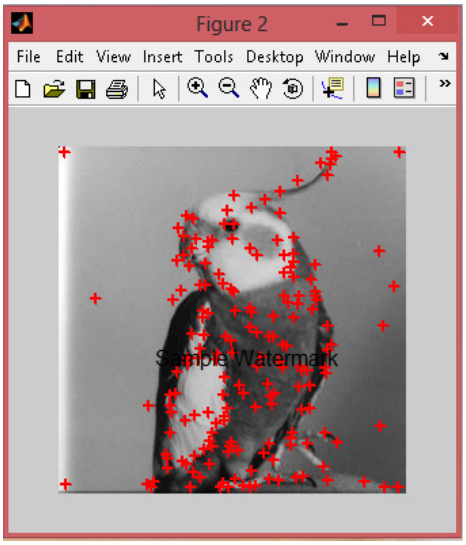


Fig 6 Result of the feature point detected with watermark on it

International Journal of Computing and Corporate Research
ISSN (Online) : 2249-054X
Volume 4 Issue 5 September 2014
International Manuscript ID : 2249054XV4I5092014-02

In fig. 5, the image of bird has been taken. The algorithm for Affine Adapted corner detection and Hessian Laplace detector has been applied on it. Firstly, the corner feature points have been detected and then watermarking is applied on it. In fig. 6, the final result of the feature point detected and watermarking applied on the feature points of the image has been shown. In both the cases the number of feature points detected have been noted down and the time taken to detect the feature points and apply watermark on it has been noted. Similarly, few attempts have been carried out for the same simulation or scenario, and then the comparison report has been carried out with the above method used.

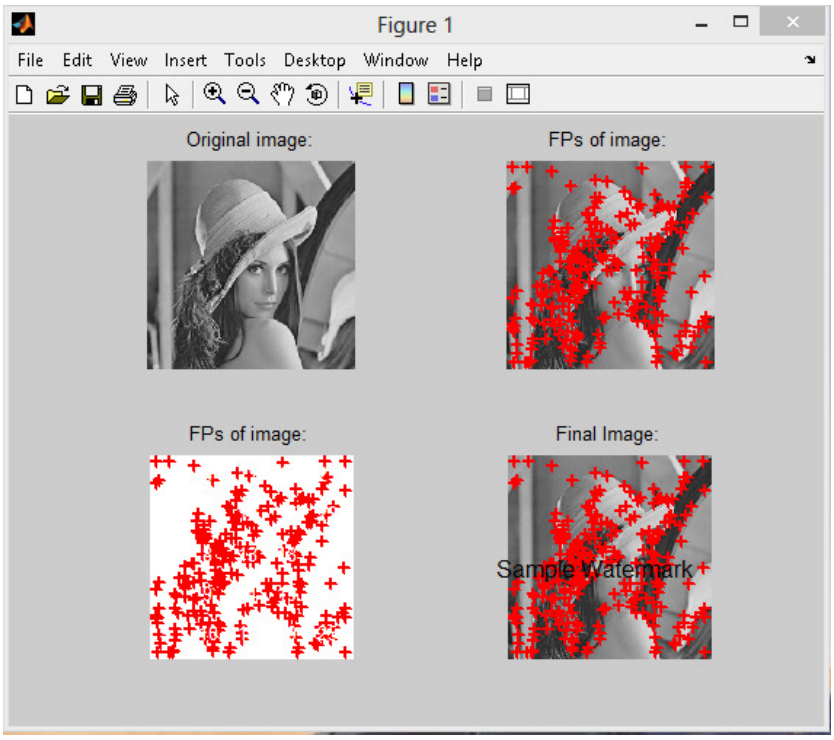


Fig 7 The feature point detection and watermarking is done on the original image

International Journal of Computing and Corporate Research
ISSN (Online) : 2249-054X
Volume 4 Issue 5 September 2014
International Manuscript ID : 2249054XV4I5092014-02

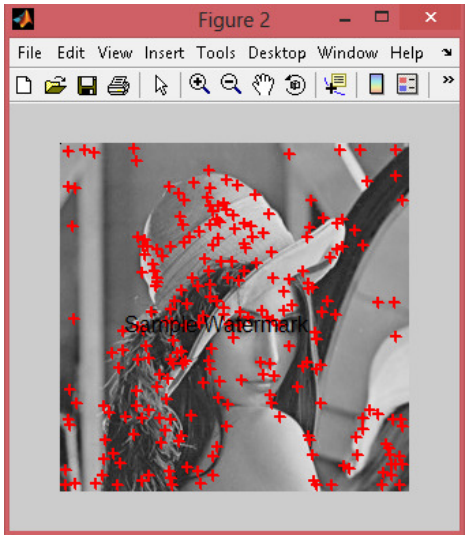


Fig. 8 Result of the feature point detection and watermark applied on it

Comparison Report

Implementation Scenario	TypeofFP	Type of Corner Detection	Number of Windows	Execution Time
1	Aff	Harris	2	5.4233223
2	HL	Affine adapted	4	4.23423

Table 1 Description of the techniques used

Implementation Attempt	Algorithm – 1 (Execution Time)	Algorithm – 2 (Execution Time)
1	4.23423	3.23423
2	6.235235	5.325235
3	6.3534523	5.35235
4	7.23235	4.5325235
5	7.3453453	5.23523124
6	6.3453	5.12354442

7	7.352353	5.343
8	8.346346346	5.23423
9	8.345346	6.34436
10	7.634634	6.345255

Table 2 Shows the comparison of the two techniques showing the time required to execute the process

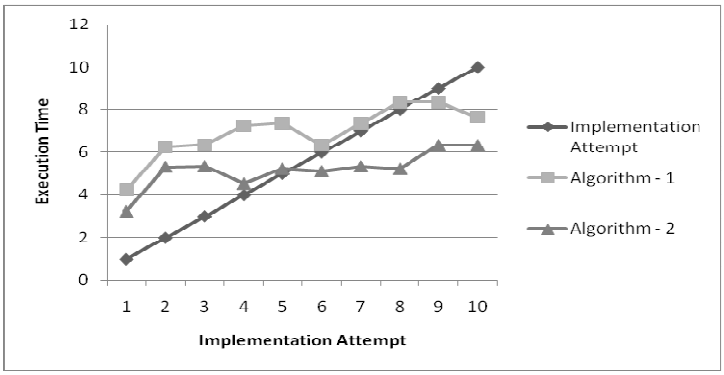


Fig 9 Graph showing the execution time required by the two techniques

Table 2, shows the comparison of the time required to detect the feature points and watermark applied on it. From the comparison report of the execution time it s noted that algorithm 2nd (Affine Adaptive method) takes less time in detecting the feature points and applying the watermark on the feature points of the image. Fig. 9 shows the graph plotted for the number of implementation attempts against the execution time taken by each attempt carried out for both the techniques.

Implementation Attempt	Points Detected : Implementation 1	Points Detected : Implementation 2
1	187	229
2	122	188
3	144	168
4	12	39
5	211	310

6	222	320
7	122	210
8	32	55
9	23	64
10	122	211

Table 3 Shows the comparison of the two techniques showing the number of feature points detected

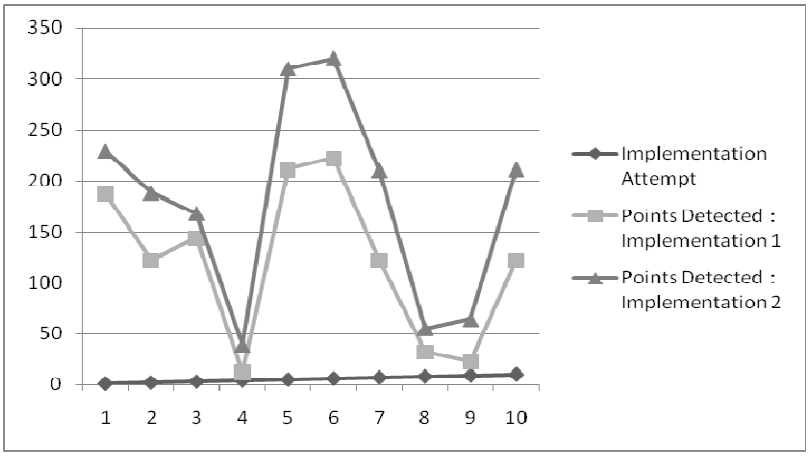


Fig 10 Graph showing the number of feture points detected by the two techniques

Conclusion

In this research work, the watermarking is applied after the corner detection of the digital images. In the future work of the research, the metaheuristic techniques including genetic algorithm, ant colony optimization or simulated annealing can be used to improve the results.

References

[1] Hal Berghel, "Watermarking Cyberspace", Comm. of the ACM, Nov.1997, Vol.40, No.11, pp.19-24

[2] G. W. Braudaway, et. al., "Protecting Publicly Available Images with a Visible Image Watermark", Proc. SPIE Conf. Optical Security and Counterfeit Deterrence Technique, Vol. SPIE-2659, pp.126-132, Feb. 1996.

[3] E. H. Adelson. Digital signal encoding and decoding apparatus. Technical Report 4,939,515, United States Patent, 1990.

[4] C.-T. Li and F.M. Yang. One-dimensional Neighborhood Forming Strategy for Fragile Watermarking. In Journal of Electronic Imaging, vol. 12, no. 2, pp. 284-291, 2003.

[5] Shapiro, Linda and George C. Stockman (2001). Computer Vision, p. 257. Prentice Books, Upper Saddle River. ISBN 0-13-030796-3.

[6] H. Moravec (1980). "Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover". Tech Report CMU-RI-TR-3 Carnegie-Mellon University, Robotics Institute.

[7] C. Harris and M. Stephens (1988). "A combined corner and edge detector". Proceedings of the 4th Alvey Vision Conference. pp. 147–151.

[8] J. Shi and C. Tomasi (June 1994). "Good Features to Track,". 9th IEEE Conference on Computer Vision and Pattern Recognition. Springer. C. Tomasi and T. Kanade (2004). "Detection and Tracking of Point Features". Pattern Recognition 37: 165–168. doi:10.1016/S0031-3203(03)00234-6.

[9] Noble (1989). Descriptions of Image Surfaces (Ph.D.). Department of Engineering Science, Oxford University. p. 45.Förstner, W; Gülch (1987 1987). "A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features". ISPRS.

[10] T. Lindeberg (1994). "Junction detection with automatic selection of detection scales and localization scales". Proc. 1st International Conference on Image Processing I. pp. 924–928.

[11] Tony Lindeberg (1998). "Feature detection with automatic scale selection". International Journal of Computer Vision 30 (2). pp. 77–116.

[12] T. Lindeberg (1994). Scale-Space Theory in Computer Vision. Springer. ISBN 0-7923-94186.

[13] T. Lindeberg (2008/2009). Scale-Space. "Wiley Encyclopedia of Computer Science and Engineering". Encyclopedia of Computer Science and Engineering (Benjamin Wah, ed), John Wiley and Sons IV: 2495–2504. doi:10.1002/9780470050118.ecse609. ISBN 0-470-05011-X.

[14] K. Mikolajczyk, K. and C. Schmid (2004). "Scale and affine invariant interest point detectors" (PDF). International Journal of Computer Vision 60 (1): pp 63–86.

doi:10.1023/B:VISI.0000027790.02288.f2.

[15] L. Kitchen and A. Rosenfeld (1982). "Gray-level corner detection". Pattern Recognition Letters 1 (2). pp. 95–102.

[16] J. J. Koenderink and W. Richards (1988). "Two-dimensional curvature operators". Journal of the Optical Society of America: Series A 5 (number 7). pp. 1136–1141.

[17] L. Bretzner and T. Lindeberg (1998). "Feature tracking with automatic selection of spatial scales". Computer Vision and Image Understanding 71,. pp. 385–392.

[18] T. Lindeberg and M.-X. Li (1997). "Segmentation and classification of edges using minimum description length approximation and complementary junction cues". Computer Vision and Image Understanding 67 (1). pp. 88–98.

[19] D. Lowe (2004). "Distinctive Image Features from Scale-Invariant Keypoints". International Journal of Computer Vision 60 (2): 91. doi:10.1023/B:VISI.0000029664.99615.94.

[20] H. Wang and M. Brady (1995). "Real-time corner detection algorithm for motion estimation". Image and Vision Computing 13 (9): 695–703. doi:10.1016/0262-8856(95)98864-P.

[21] S. M. Smith and J. M. Brady (May 1997). "SUSAN – a new approach to low level image processing". International Journal of Computer Vision 23 (1): 45–78.

[22] GB patent 2272285, list of inventors (free format), "Determining the position of edges and corners in images", published 1994-05-11, issued 1994-05-11, assigned to Secr Defence

[23] M. Trajkovic and M. Hedley (1998). "Fast corner detection". Image and Vision Computing 16 (2): 75–87. doi:10.1016/S0262-8856(97)00056-5.

[24] E. Rosten and T. Drummond (May 2006). "Machine learning for high-speed corner detection,". European Conference on Computer Vision.

[25] Leonardo Trujillo and Gustavo Olague (2008). "Automated design of image operators that detect interest points". Evolutionary Computation 16 (4): 483–507. doi:10.1162/evco.2008.16.4.483. PMID 19053496.