A Comparison between Using SIFT and SURF for Characteristic Region Based Image Steganography

Nagham Hamid[1], Abid Yahya[2], R. Badlishah Ahmad[3], and Osamah M. Al-Qershi[4]

[1,2,3]School of Communication and Computer Engineering, University of Malaysia Perlis (UniMAP)
02000 Kuala Perlis, Perlis, Malaysia

[4] School of Electrical & Electronic Engineering, University of Science Malaysia (USM)
11800 USM Pulau Pinang, Malaysia.

Abstract

Steganography is the science of invisible communication that employs different useful applications. In most of the current steganography techniques, information hiding modifies almost all the cover image, which may negatively affect the visual quality of the image and increase the possibility of losing data after the possible attacks. To solve such a problem, this paper presents a new region based steganography technique, which hides data in the robust regions of the image. Two promising approaches have been used to detect the robust regions in the image: Scale Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF). The robustness of the two algorithms has been tested against different types of attacks. Results showed that SURF based algorithm is better when detecting the robust regions correctly. Its accuracy is higher in retrieving the embedded data and that the visual quality of the embedded image is high for both algorithms.

Keywords: *Adaptive steganography; Information hiding; SIFT; SURF; Steganography.*

1. Introduction

In this modern era, computers and the internet represent the major communication media that connect different parts of the world in one global virtual world. As a result, people can easily exchange information and distance is no longer a barrier to communication. However, the safety and security of long-distance communication remains an issue. This is particularly important in the case of confidential data. The need to solve this problem has led to the development of steganography schemes. Steganography is a powerful security tool that provides a high level of security, particularly when it is combined with encryption [1].

Steganography differs from cryptography. The goal of cryptography is to secure communications by changing the data into a form that an eavesdropper cannot understand. In contrast, steganography techniques try to hide the very existence of the message itself, so that an observer does not know that it is even there. In some cases, sending encrypted information may draw the attention while invisible information will not. Accordingly, cryptography is not the best solution for secure communication; it is only part of the solution. Both sciences can be used together to protect information better. In this case, even if steganography fails, the message cannot be recovered because a cryptography technique is used as well [2].

The performance of a steganographic system can be measured using several properties. The most important property is the statistical undetectability (imperceptibility) of the data, which shows how difficult it is to determine the existence of a hidden message. Other associated measures are the steganographic capacity, which is the maximum payload that can be safely hidden in a work without producing statistically detectable objects [3], and robustness, which refers to how well the steganographic system resists the extraction of hidden data.

Almost all digital file formats can be used for steganography, but the formats that are most suitable are those that have a high degree of redundancy. The redundant bits of an object are those bits that can be changed without easily detecting the alteration. Image and audio files satisfy this requirement particularly well [4]. In fact, digital images are the most used carrier file formats owing to their popularity on the internet.

Accordingly, the present work revolves around steganography in digital images. There have been a number of image steganography algorithm proposed; these algorithms could be categorized in a number of ways [5, 6]:

- Spatial or Transform, depending on the redundancies used from either domains of the embedding process.

- Model based or Adaptive steganography if the algorithm models statistical properties before embedding and preserving them to be exploited in the embedding process.
- Active or Passive Warden, based on whether the design of embedder-detector pair takes into account the presence of an active attacker.

The majority of the existing techniques of steganography focuses on the embedding strategy and gives no consideration to the pre-processing stages. As cases in point are the encryption or data embedding based on the characteristics of the cover image. For most of the current image steganography techniques, information hiding modifies almost all the cover components, which may negatively affect the visual quality of the image and increase the possibility of losing data after the possible attacks. Adaptive steganography identifies the textural or quasi-textural areas for embedding the secret data. The latter takes statistical global features of the image before attempting to embed the secret information in particular regions of the image. These statistics will dictate where to make the changes [5].
The present paper focuses on the adaptive steganography to hide the secret information in the digital image files. Two promising approaches have been used to detect the robust regions in the image; these are Scale Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF). A comparison is presented between these techniques to find the salient regions in the image prior to the embedding process and to reveal the possible differences in their performance.

2. Overview of SIFT and SURF Techniques

In 2004, Lowe presented SIFT for extracting distinctive invariant features from images that can be invariant to image scale and rotation [7]. Then, it was widely used in image mosaic, recognition, retrieval etc [7]. In 2006, Bay et al. introduced speeded up robust features technique (SURF), and used integral images for image convolutions and Fast-Hessian detector [8]. Their experiments turned out that the latter was faster and that it worked well.
Both approaches do not only detect interest points or so called features, but also propose a method for creating an invariant descriptor. This descriptor can be used to identify the found interest points and match them even under a variety of disturbing conditions, like scale changes, rotation, changes in illumination or viewpoints or an image noise [9].
There are also many other feature detection methods, as edge detection, corner detection, etc. Different methods have their own advantages. This paper focuses on using SIFT and SURF techniques to detect the robust regions in the image. These are the characteristic regions used for information hiding.

2.1SIFT Detector

SIFT mainly includes four major stages: scale-space extrema detection, keypoint localization, orientation assignment and keypoint descriptor. The first stage used difference-of-Gaussian function (DOG) to identify the potential interest points [10], which were invariant to scale and orientation. DOG was used instead of Gaussian to improve the computation speed [10].

$$D(x, y, \sigma) = \big(G(x, y, k\sigma) - G(x, y, \sigma)\big) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (1)$$

Given a digital image $I(x, y)$, its scale space representation will be $L(x, y, \sigma)$. $G(x, y, \sigma)$ is the variable-scale Gaussian kernel with the standard deviation σ.
In the keypoint localization step, the low contrast points are rejected and the edge response is eliminated. Hessian matrix was used to compute the principal curvatures and eliminate the keypoints that have a ratio between the principal curvatures that are greater than the ratio. An orientation histogram was formed from the gradient orientations of sample points within a region around the keypoint in order to get an orientation assignment [10]. According to the paper's experiments, the best results were achieved with a 4 x 4 arrays of histograms with 8 orientation bins in each. So, the descriptor of SIFT that was used is 4 x 4 x 8 = 128 dimensions [7].
The keypoint descriptors are calculated from the local gradient orientation and magnitudes in a certain neighborhood around the identified keypoint. The gradient orientations and magnitudes are combined in a histogram representation from which the descriptor is formed [9].

2.2 SURF Detector

SURF algorithm is employed in slightly different way for detecting image features. SIFT builds an image pyramids by filtering each layer with Gaussians of increasing sigma values and taking the difference. On the other hand, SURF creates a "stack" without 2:1 down sampling for higher levels in the pyramid; a matter that results in having images of same resolution [10]. Due to the use of integral images, SURF filters the stack using a box filter approximation of second-order Gaussian partial derivatives. This is because the integral images allow the computation of rectangular box filters in a near constant time [8].
SURF has been published by Bay to tackle the problem of point and line segment correspondences between two images of the same scene or object. The latter in turn can be part of many computer vision applications. The SURF

approach can be divided into three main steps. First, keypoints are selected at distinctive locations in the image, such as corners, blobs, and T-junctions. Next, the neighborhood of every keypoint is represented by a feature vector. This descriptor has to be distinctive. At the same time, it should be robust to noise, detection errors, and geometric and photometric deformations. Finally, the descriptor vectors are matched among the different images [8]. Keypoints are found by using a so-called Fast-Hessian Detector that is based on the approximation of the Hessian matrix of a given image point. The responses to Haar wavelets are used for orientation assignment before the keypoint descriptor is formed from the wavelet responses in a certain surrounding to the keypoint [9]. Therefore, the SURF constructs a circular region around the detected key-points. Second, the SURF descriptors are constructed by extracting square regions around the key-points. Such a process results in a descriptor of sixty four-length [8].

3. Steganography Synchronization Based on Characteristic Regions

Steganography synchronization ensures that the processes of data embedding and extracting are implemented in the same region. In this paper, steganography synchronization is achieved via the characteristic regions, which can be generated using SIFT and SURF techniques, respectively. The data is embedded in particular regions in the image depending on their characteristics. The same characteristics should be used to identify the embedded regions correctly to start the extraction process. This necessitates that characteristic identification technique should be robust enough to survive after possible attacks or communication errors.

Throughout surveying the literature [11], Li et al. exploited a characteristic region, using SIFT to achieve an image watermark synchronization for copyright protection purposes. Their scheme achieved a high-capacity information hiding and generalized watermark robustness.

In the present work, SIFT and SURF are separately used in the same manner to achieve a steganography synchronization. Then, a comparison between the two techniques is presented.

3.1 Algorithm Description

The steganography synchronization algorithm consists of two stages: extracting the robust key-points in the image and data hiding in the regions centered by these key-points. The robust key-points are those points of the image that can resist a wide range of image processing operations, such as scaling and rotation. Such robust regions can be detected even when the image undergoes different attacks. The idea behind selecting those regions for hiding secret information is to make sure that the locations of the regions in which the data is hidden can be identified without an embedding map. Besides, the regions in which the data is embedded are not fixed and highly dependent on the characteristics of the image used as a cover. In addition, selecting a few regions for hiding data will minimize the distortion of the stego-image. In the data hiding stage, the secret information is embedded using a DWT-based technique. The DWT-based techniques are proven to be more robust compared to other techniques, like Discrete Fourier Transform (DFT) or Discrete Cosine Transform (DCT). In the next section, the two stages are described.

3.2 Extracting Key-points

After applying SIFT or SURF on the cover image, the extracted key-points are presented using three parameters: coordinates, scale, and orientation. The coordinates of the key-points are the coordinates of the circular regions, and of radius r, in which the secret data will be embedded. When SIFT is used for extracting the key-points, Li et al. suggested that the scale of a key-point should be between 4 and 8 for the best results. These values can define about 5–10 key-points for an efficient watermark synchronization for common images. If the circular regions are generated directly following the above procedures, some of them may overlap with the others. To avoid that problem, the regions should be disjoint. If two regions overlap, only the one that corresponds to a bigger scale is selected as it has a better stability. Fig. 1 shows an example of the characteristic regions generated on Lena's image.

Fig. 1 Characteristic regions extracted from Lena's image [11].

In the same manner, when SURF is used to extract the key-points, some points will not be used in order to avoid

any intersected regions due to having very close key-points. To guarantee that the local regions are disjoint, the extracted local regions are first sorted based in their scales on a descending order. Then, each point is considered by calculating the Euclidian distance d between the selected points and all other points in the list. All d values should be greater than ($2 \cdot r$), where r is the radius of the local region.

3.3 Data Embedding and Extracting

After extracting several invariant circular regions for steganography synchronization, the secret data can be embedded into the selected regions. It should be noted that due to the discrete property of digital images, the local regions that can be actually used is not circular but square. As a result, the bordering area of a circular region is first padded with zeros to construct a square region. Then, the information will be embedded, and zero-removal is employed to obtain the stego-circular region. Fig. 2 shows the detailed steps.

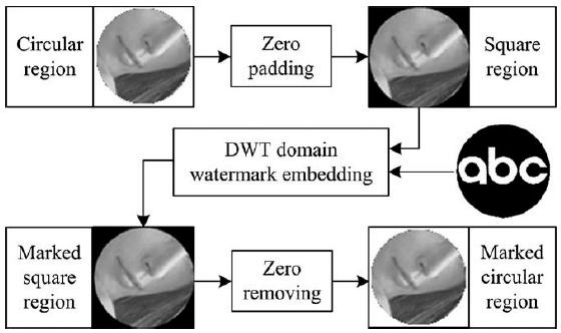

Fig. 2 Zero-padding and zero-removal [11].

The information is embedded in Discrete Wavelet Transform (DWT) domain in a content-based manner. For each characteristic region, one level DWT is applied to produce the wavelet coefficients, using the 9/7 biorthogonal wavelet, as shown in Fig. 3. To embed a secret bit b, the corresponding horizontal and vertical wavelet coefficients are first selected and denoted by $H(x, y)$ and $V(x, y)$, respectively.

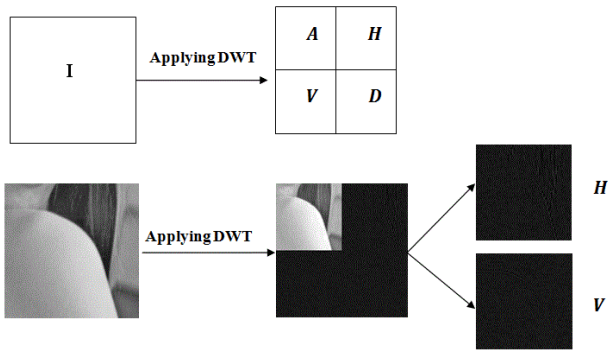

Fig. 3 Decomposing image into 4 sub-bands using DWT.

Then, b is embedded by increasing the difference between $H(x, y)$ and $V(x, y)$. The rules of wavelet coefficient modification are as follows.
If $b = 1$ and $D_1 = H(x, y) - V(x, y) < T$ (T is a threshold to control information invisibility), $H(x, y)$ will be increased while $V(x, y)$ will be decreased by inserting the secret message.

$$\begin{cases} H'(x, y) = H(x, y) + \frac{T - D_1}{2} \\ V'(x, y) = V(x, y) - \frac{T - D_1}{2} \end{cases} \quad (2)$$

Else if $D_1 = H(x, y) - V(x, y) \geq T$, do nothing;

If $b = 0$ and $D_2 = V(x, y) - H(x, y) < T$, the same process is implemented.

$$\begin{cases} H'(x, y) = H(x, y) - \frac{T - D_2}{2} \\ V'(x, y) = V(x, y) + \frac{T - D_2}{2} \end{cases} \quad (3)$$

Else if $D_2 = V(x, y) - H(x, y) \geq T$, do nothing.

Finally, one level inverse DWT is applied to obtain the stego-region. The extraction phase starts with the same steps of extracting the key-points and the characteristics regions. After that, one level DWT is applied to each characteristic region to obtain the wavelet coefficients. The horizontal and vertical coefficients are determined and denoted by $H(x, y)$ and $V(x, y)$, respectively. Then, each bit b can be extracted by comparing the corresponding horizontal and vertical coefficients, as shown in Eq. (4).

$$b = \begin{cases} 1 \ if \ H(x, y) > V(x, y) \\ 0 \ if \ V(x, y) > H(x, y) \end{cases} \quad (4)$$

3.4 Embedding and Extracting Procedures

The detailed data embedding procedures are as given below:

1. The characteristic regions are extracted from the cover image using SURF or SIFT. Then, the resultant invariant points are examined to avoid any intersected regions with $r = 64$. Some points are eliminated throughout this step.
2. Using the final list of points, the embedding regions are located in the cover image as circular regions of a radius r.
3. For each embedding region, one level DWT on each characteristic region is applied to produce the wavelet coefficients. In our algorithm, the 9/7 biorthogonal wavelet is adopted.
4. Horizontal and vertical high frequency coefficients are scanned in a raster way, and the data bits are embedded by modifying the horizontal and vertical coefficients in a content-based manner, as in the SIFT based scheme.
5. Finally, one level inverse DWT is applied to obtain the stego region, and then the original characteristic region is replaced with the stego one. The whole embedding phase is illustrated in Figure 4. For the entire extracted characteristic region, the aforementioned embedding procedures are conducted repeatedly to produce the whole stego-image.

The first two steps of data extraction phase are exactly the same as data embedding. Characteristic regions are first extracted from the possibly distorted image, using SURF or SIFT techniques. The invariant key-points are examined to avoid any intersected regions. Then, the embedding regions are determined. Later, a payload extraction is done on each local region, as given in Eq. (4).

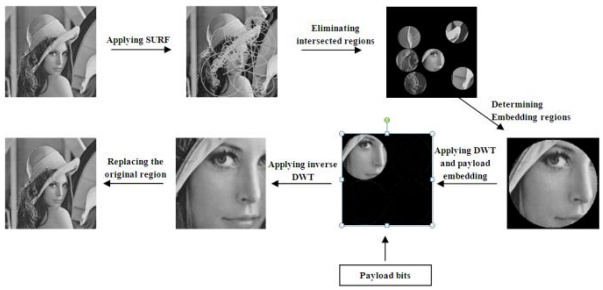

Fig. 4 The embedding phase

4. Experimental Results

In order to compare between exploiting SIFT and SURF in steganography synchronization, three standard gray images of the size (512x512) pixels have been used. The radius of the circular characteristic regions is set to 64 pixels in both cases. The embedding and extracting process have been repeated 100 times using randomly generated data bits. For comparison purpose, 1- level and 2- level of 9/7 biorthogonal wavelet have been used. The threshold T used for payload embedding is set to 1, which is determined experimentally.

To test the robustness of the proposed scheme, different attacks of different levels are applied to the stego-image image. The attacks which have been involved are JPEG compression, Gaussian Additive noise, median filter, and low pass filter.

For the purpose of evaluation, the attacks are applied to the stego-image, the extracted payload is compared with the embedded payload and the Bit Error Rate (BER) is calculated using Eq. (5).

$$BER = \frac{number\ of\ error\ bits}{total\ number\ of\ embedded\ bits} \times 100 \qquad (5)$$

Beside the BER, the accuracy of synchronization (accuracy of the correct detection of the characteristic region, denoted by (ADR) using SIFT and SURF) is measured, by calculating the percentage of the number of regions that have been correctly identified during the extraction phase.

For each type of attacks, the process is repeated 100 times and the averages are calculated as given in Tables 1, 2, and 3. Another comparison is presented in Table 4 between exploiting 1-level DWT and 2-level DWT in terms of the hiding capacity and the visual quality of the stego-image. The capacity is measured by calculating the number of payload bits that can be embedded in the image while the visual quality is measured by taking into account the Peak Signal to Noise Ratio (PSNR), as given in Eq. (6).

$$PSNR(I, I_s) = 10 \cdot \log_{10} \frac{MAX_I^2}{\frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left\|I(i,j) - Is(i,j)\right\|^2} \qquad (6)$$

Where I is the original image; Is is the stego-image; MAX_I is the maximum possible pixel value of the image I [12].


Lena


Bridge


Peppers

Fig. 5 Standard test images used for evaluating

Table 1: A comparison between SFIT and SURF using 'Lena's image'

Type of Attack	1-level DWT				2-level DWT			
	SIFT		SURF		SIFT		SURF	
	ADR (%)	BER (%)	ADR (%)	BER (%)	ADR (%)	BER (%)	ADR (%)	BER (%)
No Attack	87.4	6.40	100	0	83.80	5.52	97.33	0.94
Gaussian Noise (25dB)	12.2	43.36	73.67	37.97	14.40	40.80	72.83	30.25
Gaussian Noise (35dB)	32.8	36.21	98.17	21.95	35.60	27.73	92.83	10.55
Gaussian Noise (45dB)	60.6	20.10	100	2.86	58.20	14.78	96.33	1.37
JPEG compression (Q=100%)	84.2	8.02	100	0.07	78.00	7.54	97.00	1.05
JPEG compression (Q=90%)	41.4	39.67	100	31.03	42.20	23.44	96.50	3.70
JPEG compression (Q=80%)	37.6	44.74	99.83	39.84	45.80	26.99	95.67	13.66
Median Filter (3x3)	34.2	47.29	83.33	47.02	40.60	32.66	83.33	23.21
Low pass filter (3x3)	48.4	48.36	100	49.17	53.00	25.99	99.33	18.88

Table 2: A comparison between SFIT and SURF using image 'Bridge'

Type of Attack	1-level DWT				2-level DWT			
	SIFT		SURF		SIFT		SURF	
	ADR (%)	BER (%)	ADR (%)	BER (%)	ADR (%)	BER (%)	ADR (%)	BER (%)
No Attack	84.67	7.79	84.5	7.92	86.67	5.78	88.33	4.85
Gaussian Noise (25dB)	40	42.93	85.17	35.14	37.33	37.57	81.00	27.06
Gaussian Noise (35dB)	56.67	32.11	84.67	23.01	57.33	23.41	88.17	11.30
Gaussian Noise (45dB)	76	14.00	83.83	10.27	73.33	11.91	88.83	4.68
JPEG compression (Q=100%)	76.33	11.98	84.83	7.81	81.00	8.26	88.83	4.68
JPEG compression (Q=90%)	61.33	34.92	85.17	29.78	68.67	16.15	88.50	7.01
JPEG compression (Q=80%)	59.67	41.57	85.50	37.19	54.00	27.70	90.00	14.77
Median Filter (3x3)	41.33	49.39	41.67	48.58	41.67	40.54	39.67	39.07
Low pass filter (3x3)	41.33	49.39	100	48.56	65.67	36.17	98.33	29.21

Table 3: A comparison between SFIT and SURF using the image of 'Peppers'

Type of Attack	1-level DWT				2-level DWT			
	SIFT		SURF		SIFT		SURF	
	ADR (%)	BER (%)	ADR (%)	BER (%)	ADR (%)	BER (%)	ADR (%)	BER (%)
No Attack	91.5	4.74	100	0.28	90.75	4.48	98.25	1.35
Gaussian Noise (25dB)	45.75	44.08	88.25	39.68	43.25	40.52	88.00	31.55
Gaussian Noise (35dB)	66	30.88	97.00	22.14	65.25	22.33	94.50	10.87
Gaussian Noise (45dB)	77.25	13.90	100	3.02	73.75	12.60	97.00	1.80
JPEG compression (Q=100%)	89	6.05	100	0.36	85.50	6.70	98.00	1.42
JPEG compression (Q=90%)	76.25	34.87	100	31.44	72.50	15.49	98.25	3.90
JPEG compression (Q=80%)	75.75	41.66	91.25	40.57	69.25	22.66	84.00	16.67
Median Filter (3x3)	69.75	49.01	75.75	49.43	50.00	34.54	75.00	22.92
Low pass filter (3x3)	52.5	51.49	100	53.32	82.25	27.85	100	16.51

Table 4: A comparison between SIFT and SURF in terms of PSNR and hiding capacity

Number of key-points	DWT Levels	SIFT			SURF		
		Lena	Bridge	Peppers	Lena	Bridge	Peppers
		5	3	4	6	6	4
Payload (bits)	1-level DWT	13,800	7,848	10,464	15,696	15,696	10,464
	2-level DWT	2,580	1,548	2,064	3,096	3,096	2,064
PSNR (dB)	1-level DWT	46.57	43.60	48.32	45.28	39.87	48.12
	2-level DWT	44.25	43.49	47.37	41.53	39.19	47.62

5. Discussion and Conclusion

The aim of this paper is to compare between exploiting SURF and SIFT in steganography synchronization. For this purpose, each technique has been combined with a DWT based data hiding method and the two resultant schemes have been tested on the same test images. The experiment results in Tables 1, 2, and 3 demonstrate the advantages of using SURF as it shows a higher robustness indicated by the lower BER values.

Clearly, the robustness of the SURF-based scheme increases when 2-level DWT is used for hiding data; especially against JPEG compression. However, the median and the low pass filters are still very challenging.

Utilizing higher levels of DWT is useful for enhancing the robustness. However, it has a negative effect on the visual quality, in terms of PSNR, as shown in Table 4. Besides, the higher levels of DWT affects the ability of SURF and SIFT to extract correctly the key-points. This is because higher levels of DWT result in higher levels of image degradation. Nevertheless, the visual quality of the stego-images is still high as the PSNR values are in an acceptable range.

The hiding capacity, which can be achieved, is relatively limited; a matter which makes the proposed scheme more appropriate for copyright protection applications. In order to use this algorithm in transmitting secret data of a bigger size, the data among several images must be divided. For a feature work, it is expected to enhance the proposed scheme in terms of increasing the hiding capacity and robustness as well. This may be achieved by adopting different frequency domain-based data hiding techniques. Moreover, more possible attacks should be investigated.

REFERENCES

[1] S.A. Halim and M.F.A. Sani, "Embedding Using Spread Spectrum Image Steganography With Gf (2m)", in Proc. of the 6th IMT-GT Conference on Mathematics, Statistics and its Applications (ICMSA), University Tunku Abdul Rahman, Malaysia, Nov. 3-4 2010, pp. 659 - 666.

[2] N.N. EL-Emam, "Hiding A Large Amount Of Data With High Security Using Steganography Algorithm", Journal of Computer Science, Vol. 3, No. 4, 2007, pp. 223-232.

[3] I.J. Cox, M.L. Bloom, J.A. Fridrich, and T. Kalkert, Digital watermarking and steganography, 2nd ed., Morgan Kaufman Publishers, USA, 2008.

[4] T. Morkel, J.H.P. Eloff, and M.S. Oliver, "An overview of image steganography", in Proc. ISSA. Sandton, South Africa, June/July 2005, pp. 1-11.

[5] A. Cheddad, J. Condell, K. Curran, and M. Kevitt, "Digital image steganography: Survey and analysis of current methods", Signal Processing J. Vol.90, No. 3, 2010, pp. 727–752.

[6] M. Kharrazi, H. T. Sencar, and N. Memon, "Image Steganography: Concepts and Practice", WSPC/Lecture Notes Series: 9in x 6in, 2004, pp. 1-49.

[7] D. Lowe. "Distinctive Image Features from Scale-Invariant Keypoints", IJCV, Vol. 60, No. 2, 2004, pp. 91–110.

[8] H. Bay, "From wide-baseline point and line correspondences to 3D", Ph.D. thesis, Swiss Federal Institute of Technology, Switzerland, 2006.

[9] J. Bauer, N. S¨underhauf, and P. Protzel, "Comparing several implementations of two recently published feature detectors", in Proc. of the International Conference on Intelligent and Autonomous Systems, IAV, France, 2007, pp. 1-6.

[10] L. Juan and O. Gwun, "A comparison of SIFT, PCA-SIFT and SURF", International Journal of Image Processing (IJIP), Vol. 3, No. 4, 2009, pp.143-152.

[11] L. Li, J. Qian, J.-S. Pan, "Characteristic region based watermark embedding with RST invariance and high capacity", International Journal of Electronics and Communications, Vol. 65, No. 5, 2011, pp. 435-442.

[12] H.S. Majunatha Reddy, K.B. Raja, "High capacity and security steganography using discrete wavelet transform", IJCSS, Vol. 3, No. 6, 2010, pp. 462-47.

Nagham Hamid awarded her B.Sc. degree in Electronic and Communication Engineering from Al-Nahrain University (Saddam University previously), Baghdad, Iraq in 1999. She obtained her M.Sc. degree in Modern Communication Engineering, in 2002, from the same university. Currently, she is a Ph.D. student in University Malaysia Perlis (UniMAP), at the School of Computer and Communication Engineering. Her research interests are on communication engineering, information technology, digital signal processing, and image based steganographic techniques.

Abid Yahya awarded his B.Sc. degree from the University of Engineering and Technology, Peshawar, Pakistan in Electrical and Electronic Engineering majoring in telecommunication. Dr. Abid Yahya began his career on a path that is rare among other Researcher executives and awarded his M.Sc and Ph.D. degree in Wireless & Mobile systems, in 2007 and 2010 respectively, from the Universiti Sains Malaysia, Malaysia. Currently, he is working at the School of Computer and Communication Engineering, Universiti Malaysia Perlis (UniMAP). His professional career outside of academia includes writing for the International Magazines, Newspapers as well as a considerable career in freelance journalism. He has applied this combination of practical and academic experience to a variety of consultancies for major corporations.

R.Badlisha Ahmad He obtained his Bachelor degree in Electrical & Electronic Engineering from Glasgow University in 1994. He obtained his M.Sc and PhD in 1995 and 2000, respectively from the University of Strathclyde, UK. His research interests are on computer and telecommunication network modeling using discrete event simulators, optical networking & coding and embedded system based on GNU/Linux for vision. He has five years teaching experience in Universiti Sains Malaysia. Since 2004 until now he has been working with Universiti Malaysia Perlis (UniMAP). Currently, he is the Dean at the School of Computer and Communication Engineering and the Head of Embedded Computing Research Cluster.

Osamah M. Al-Qershi received his Bachelor of Science (B.Sc. degree) in computer control engineering from the University of Technology, Baghdad, Iraq in 1998, and M.Sc. degree in image processing form Universiti Sains Malaysia (University of Science, Malaysia) in 2011. Currently, he is a Ph.D. student at the School of Electrical & Electronic Engineering in Universiti Sains Malaysia (USM). His research interest is in the area of digital image watermarking and forensics.