SPATIAL HASHING SCHEME FOR IMAGE AUTHENTICATION

Sedki B. T. Younis
sedkibakir@yahoo.com

Omar M. Ahmed
al_saydia1@yahoo.com

University of Mosul
College of Electronics Engineering
Computer and Information Engineering Department
Mosul-Iraq

ABSTRACT

Hashing techniques have been used extensively in applications such as content authentication, database search and watermarking. In this paper, we propose a spatial hashing scheme to provide a compact representation of an image that can be used for Image authentication. The proposed hashing scheme extracts the image feature based on pixel proximities in the spatial domain. Moreover, the proposed scheme uses a secret keys to modulate and permutate the pixels and final hash values. The robustness of the proposed technique has been tested with standard benchmark attacks.

Keywords: *Image Authentication, Spatial Hashing,*

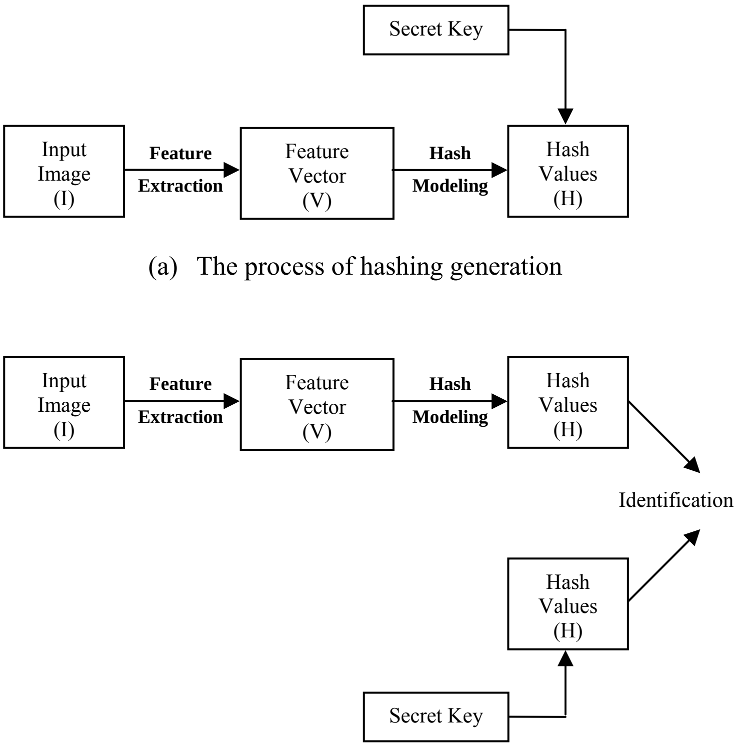
I. INTRODUCTION

Due to the popularity of digital technology, more and more digital images are being created and stored every day. This introduces a problem for managing image databases. One cannot determine if an image already exists in a database without exhaustively searching through all the entries. Further complication arises from the fact that two images appearing identical to the human eye may have distinct digital representations, making it difficult to compare a pair of images. Given a suitable algorithm to generate image identifiers, or an *image hash function*, one may use standard algorithms that search and sort n binary strings in time proportional to log n rather than to n [1].

Other applications of image hashing lie in the area of image authentication. The reliable identification or authentication of content is popular in the process of storing and distribution of digital information. In cryptography, hash functions are typically used for digital signatures to authenticate the message being sent so that the recipient can verify its source. A key feature of conventional hashing algorithms such as MD5 and SHA-1 is that they are extremely sensitive to the message, i.e. a one bit change in the input changes the output dramatically [2].

However, most cryptographic hashes are sensitive and meet difficulties when coming to multimedia contents which has different representation versions, such as format conversion and compression. A multimedia hash functions takes into account changes in the visual domain. In particular, a perceptual multimedia hash is required to be invariant under image manipulations that do not alter the image appearance significantly. Moreover, technologies for the identification of multimedia content are also included in the upcoming and ongoing standardization activities such as MPEG-7 or MPEG-21. Considering about its application scenarios, perceptual hashing function should satisfy the following requirements: ability of discrimination, robustness against attacks or modifications, dimension of perceptual hash, complexity and performance of the hash calculation, complexity and performance of the hash retrieval and algorithm security.

A general scheme of the perceptual hashing technology is shown in Fig. 1. The most challenging part of this scheme is to extract the feature vector which has to represent the image and yet robust to various distortions. Normally, image features that are invariant to allowed content-preserving image processing operations are identified and used to generate the hash functions [2, 3].



(a) The process of hashing generation

(b) The process of image authentication

Fig. 1 Image Hashing and Authentication

II. PROPOSED HASHING SCHEME

Some of the methods to extract hash values of an image that have been proposed in the literature are discussed. In [1], Venkatesan presented an image hashing technique, which divided a wavelet transformed image into tiles and extract the statistical features of tiles as hashes. In [2], Monga presented a perceptual image hashing technique using feature points, where the wavelet transform based on an end-stopped wavelet is used to feature extraction. In [3], Yang compared four normalized block mean based image perceptual hashing functions. The first two methods are based on normal block mean values of original content while the third and forth methods combined simplified Radon transform, which enhanced the rotation attacks. In [4], Ahmed, proposed a secure image hashing scheme that allows acceptable manipulations like JPEG compression and low pass filtering and is sensitive to enough in detecting malicious manipulations. In [5], Swaminathan developed an algorithm for generating an image hash based on Fourier transform and controlled randomization. The proposed hash function is resilient to content-preserving modifications, such as moderate geometric and filtering distortions. In this paper, we propose a spatial hashing scheme for image authentication. The proposed hash generation technique is discussed below:

A. *Hash generation*

In this section we describe the proposed hashing scheme used for image authentication. The hash generation structure is based on the proximity values of pixels in the spatial domain. A block diagram for the hash generation module is shown in Fig. 2. The steps involved in the hash generation process is:

1) The input image I of dimension N x N pixels is partitioned into non-overlapping blocks, each of dimensions P x P pixels. This gives a total of N^2/P^2 blocks. We represent each block by B_i , where i =1,…,N^2/P^2. Let $B_i\ (x, y)$ represent the gray value of a pixel at spatial location (x, y) in the block B_i .

2) Let K_1 be a secret key which used to generate a random block W_i of size P x P pixels. Using the key K_1, a random intensity transformation is applied to each block by modulating each pixel of B_i with a random block W_i generated using the key K_1 to get intensity-transformed blocks. Then the modulated block is permuted using another key K_2 to get a modulated and permutated block , $\overline{B_i}$ as shown below:

$$\overline{B_i}=Permute_{K2}\,(\,B_i(\,x,y)W_i(\,K_1\,)\,) \qquad (1)$$

3) For each one of the permutated and modulated block ($\overline{B_i}$), the Vertical-Mean (VM) and Horizontal-Mean (HM) are obtained by calculating the mean for each row and coulomb of the block. This results a two vectors named Vertical-Mean (VM) and Horizontal-Mean (HM). Then the main spatial features are extracted by accumulating the differences between the mean vectors. The Horizontal-Proximity (HP) and Vertical-Proximity (VP) for each block is calculated as follows:

$$VP = \sum_{i=1}^{P-1} Difference(VM[i+1], VM[i]) \qquad (2)$$

$$HP = \sum_{i=1}^{P-1} Difference(HM[i+1], HM[i]) \qquad (3)$$

The Intermediate hash vector $H_{\mathrm{intermediate}}$ is formed which is a set of vectors containing VP and HP of all the blocks of I as follows:

$$H_{intermediate}=\{\,(VP_1, HP_1)\,,\ldots,\,(VP_{N^2/P^2}, HP_{N^2/P^2})\,\} \qquad (4)$$

4) Obtain the median value (M_d) for the H $_{\mathrm{intermediate}}$ vector

$$M_d=median\,(\,H_{intermediate}\,) \qquad (5)$$

5) Normalize the intermediate hash sequence (H $_{\mathrm{intermediate}}$) in to binary form and obtain the hash values as:

$$H_{binary} = \begin{cases} 0 & if \quad H_{\mathrm{int}\,ermediate}(i) < M_d \\ 1 & if \quad H_{\mathrm{int}\,ermediate}(i) \geq M_d \end{cases} \qquad (6)$$

5) The final hash H_{Final} is formed by permuting the entries of the binary hash (H $_{\mathrm{binary}}$) hash with another secret key, The reason for permuting the hash entries is to prevent an attacker from knowing the transformed hash values of each image block.

$$H_{Final}=Permute_{K3}\,(\,H_{intermediate}\,) \qquad (7)$$

B. *Image Authentication*

A block diagram for image authentication module is shown in Fig. 3. Description of the steps involved in authentication process are shown below:

1) The received image $\hat{I}$ of dimension N x N pixels is processed through the same steps 1-5 as outlined in Section II-A to calculate the hash H_{Final} .
2) To determine whether the received image is authentic or unauthentic, we choose the hamming distance between the binary hashes, normalized it with respect to the length (L) of the hash as a performance metrics. The normalized Hamming distance is defined as:

$$d(h_1,h_2) = \frac{1}{L}\sum_{k=1}^{L} |h_1(k) - h_2(k)| \qquad (8)$$

Which expected to be close to 0 for similar images and close to 0.5 for dissimilar ones. As more parts of a picture are changed, the manipulated image and the original image becomes more dissimilar. For an ideal hashing scheme, the normalized Hamming distance between the corresponding hashes should increase accordingly [5].

III. EXPERIMENTAL RESULTS

In this section, we present a number of experimental results to test the robustness and discriminative capability of the proposed hashing scheme. In our experiments, we have used images of size 256 × 256 pixels. The image is divided in to non-overlapping blocks of size 8 × 8. This gives a total of 256 blocks. After modulation and permutation of block pixels, the VP and HP features is calculated. The VP and HP is quantized to formulate the intermediate hash values. Finally, the features vector is converted to binary and permutated. The size of the final hash is 2048 bits.

To demonstrate the robustness against acceptable manipulations, we test our scheme against JPEG compression, rotation, wiener filtering, median filtering, average filtering and cropping effect. We measure the normalized Hamming distance between the hashes of the original images and manipulated images. We compare the results of the proposed hashing scheme with block mean scheme discussed in [3].

The comparison results in terms of normalized hamming distance for the proposed scheme compared with the block mean scheme under different content-preserving manipulations is shown in Fig. 4-9. It is clear for the figures that the proposed scheme has a good robustness compared with block mean scheme against different manipulations presented in this work.

In most applications, the process of image authentication is similar to a hypothesis testing process with the following two hypothesizes:

H_0: Image is not authentic
H_1: image is authentic

The ROC curve characterizes the receiver's performance by classifying the received signal into one of the hypothesis states. For each original image, we compute and store the hash values, which we denote as h_1. Given the received image, we find its hash value and declare it to be authentic if the normalized Hamming distance between the hashes satisfies $d(h_1, h_2) < \eta$ where η is a decision threshold. We record the number that are correctly classified as authentic to give us an estimate of the probability of correct detection (PD). For a given η, we also record the number of processed versions of other images that are falsely classified as original image and obtain an estimate of the probability of false alarm (PF). We repeat this process for different decision thresholds , and arrive at the ROC [3, 5].

The ROC curves are shown in Fig. 10-11 for the Wiener and Median filtering operations. We obtain the ROC curve by adjusting the threshold and record the number of PD and PF. It is clear that the proposed techniques has better discrimination capability compared with block mean technique.

IV. SECURITY

There are many security issues that have to be considered for an image hashing scheme. Given the hash generation algorithm and the image, an attacker should not be able to predict the image hash. This means that the hash generation process should be *key-dependent*. Furthermore, if the image hash is exposed, it should be extremely difficult to derive the secret key that was used to generate the hash. Another attack that can be launched is to defeat the authentication process. By defeating the authentication process we mean that the attacker manipulates the image in such a way that the manipulated image and the actual image are visually different, however, the hashes of both the images are same or within the authentication bound [6].

The security for the proposed technique is achieved by using three secret keys. The K_1 key is used to generate a random pattern in order to be modulated with the pixels block and K_2 key is used to permutate the pixels before feature extraction stage such that the attacker can't create a forged image such that the hash of the forged image match with the hash of the original image. Moreover, another key K_3 is used to permutate the final hash making it difficult for an attacker to estimate the values of the hash for the image.

V. CONCLUSION

In this paper, we have proposed an image hashing scheme based spatial domain for feature extraction to produce a compact binary hash for image authentication purposes. The algorithm is tested to show it robustness against common signal processing operations. Experimental results showed that the proposed technique is robust against acceptable manipulations like JPEG compression, geometric and filtering operations. Moreover, the proposed algorithm is based on key-dependent for feature extraction and permutation of the pixels and final hash values.

REFERENCES

[1] Ramarathnam Venkatesan, S.-M. Koon, Mariusz H. Jakubowski, and P. Moulin, ***"Robust Image Hashing"*** IEEE Int. Conf. on Image Processing: ICIP 2000, Vancouver (BC), CA, Sep. 2000.

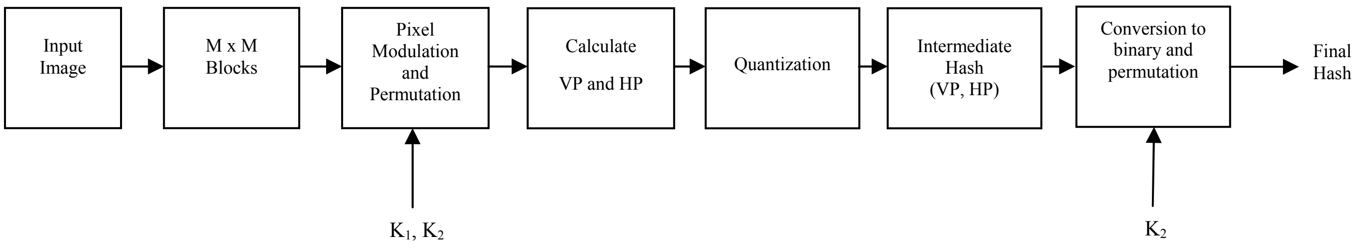
[2] *Vishal Monga* and *Brian L. Evans*, **"Robust Perceptual Image Hashing Using Feature Points"**, *Proc. IEEE Int. Conf. on Image Processing*, Oct. 24-27, 2004, vol. 3, pp. 677-680, Singapore.

[3] B. Yang, F. Gu and X. Xiu, ***"Block Mean Value Based Image Perceptual Hashing for Content Identification"***, IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Pasadena, USA, 2006.

[4] Ahmed F. and Siyal, M.Y. , ***"A Secure and Robust Hashing Scheme for Image Authentication"***, Fifth International Conference on Information, Communications and Signal Processing, Dec. 2005.

[5] A. Swaminathan, Y. Mao and M. Wu, ***"Robust and Secure Image Hashing"***, *IEEE Transactions on Information Forensics and Security*, vol. 1, No. 2, June 2006.

[6] F. Ahmed and M. Siyal, ***"A Novel Hashing Scheme for Image Authentication"***, Innovations in Information Technology, Dubai, Nov. 2006.



Input Image
M x M Blocks
Pixel Modulation and Permutation
Calculate VP and HP
Quantization
Intermediate Hash (VP, HP)
Conversion to binary and permutation
Final Hash
K_1, K_2
K_2

Fig. 2 Block diagram for Hash Generation Module

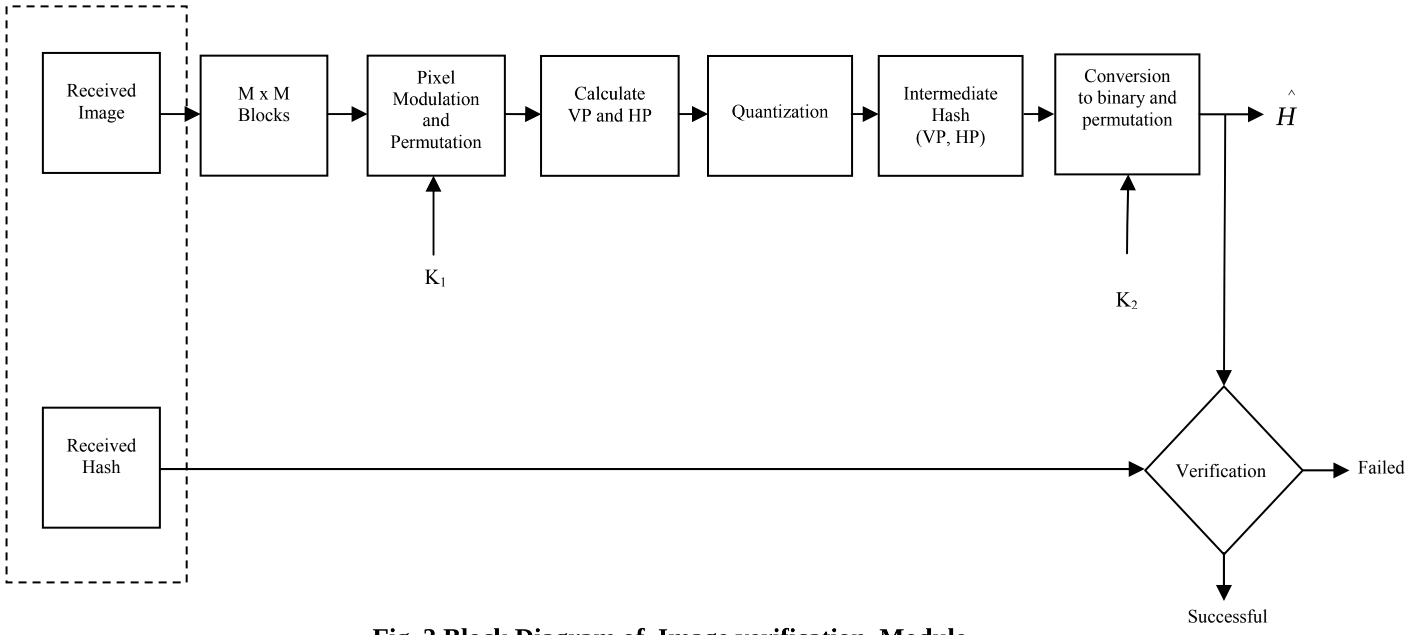
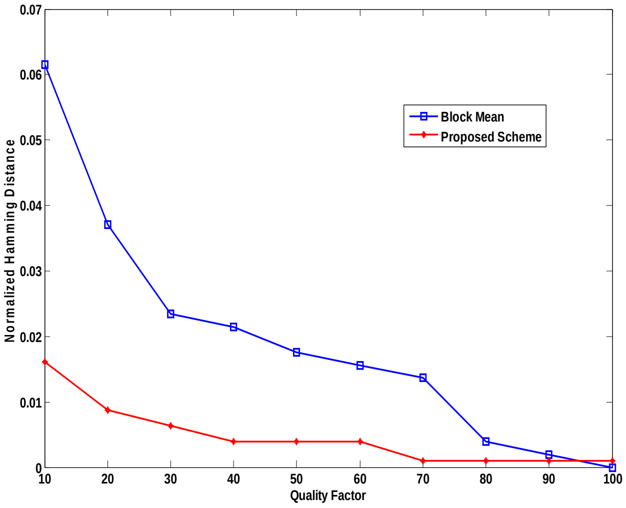


Fig. 3 Block Diagram of Image verification Module



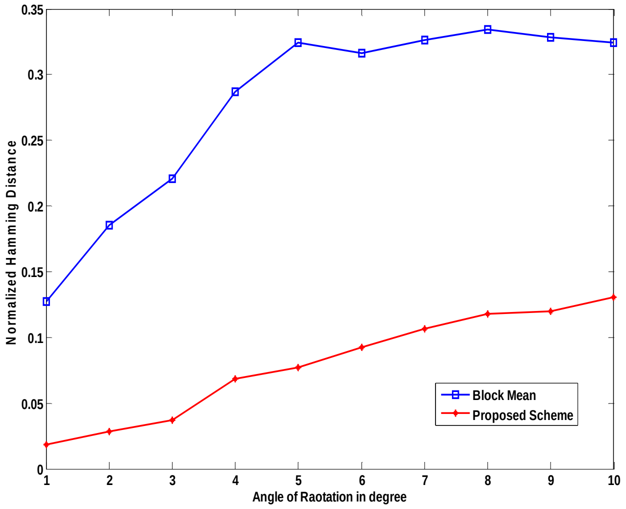


Fig. 4 The Effects of JPEG Compression

Fig. 5 The Effect of Rotation

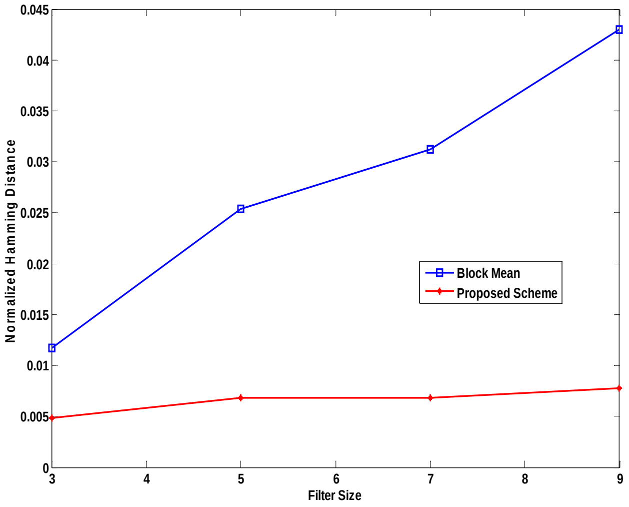


Fig. 6 The Effect of Wiener Filtering

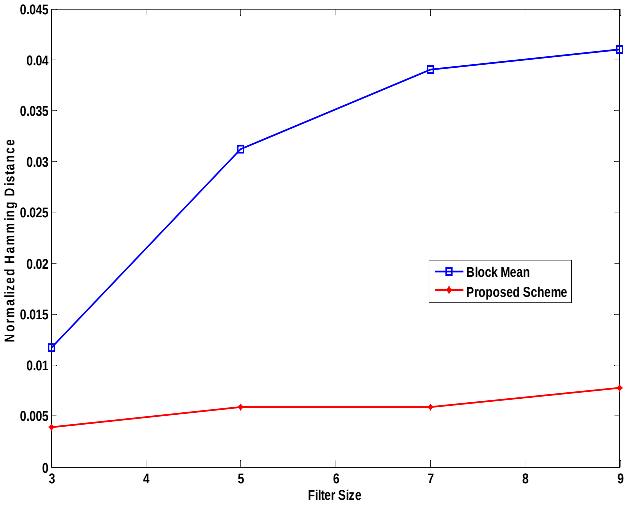


Fig. 7 The Effect of Median Filtering

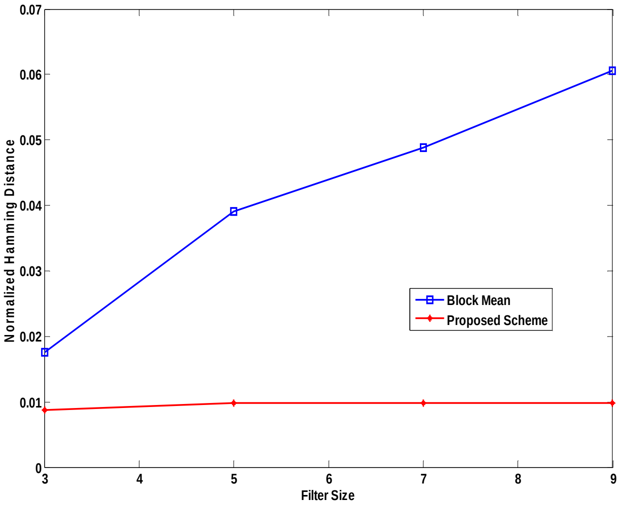


Fig. 8 The Effect of Average Filtering

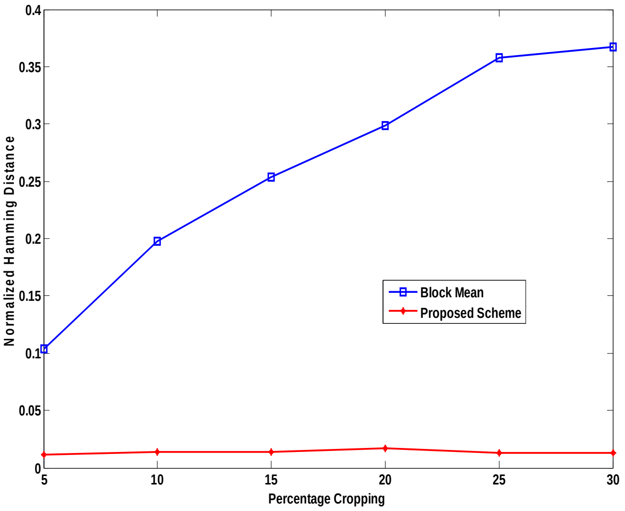


Fig. 9 The Effect of Cropping

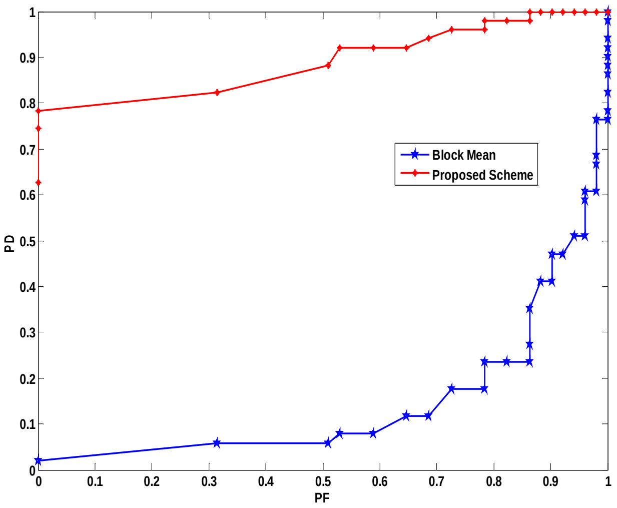


Fig. 10 ROC curve for the Wiener Filtering Effects

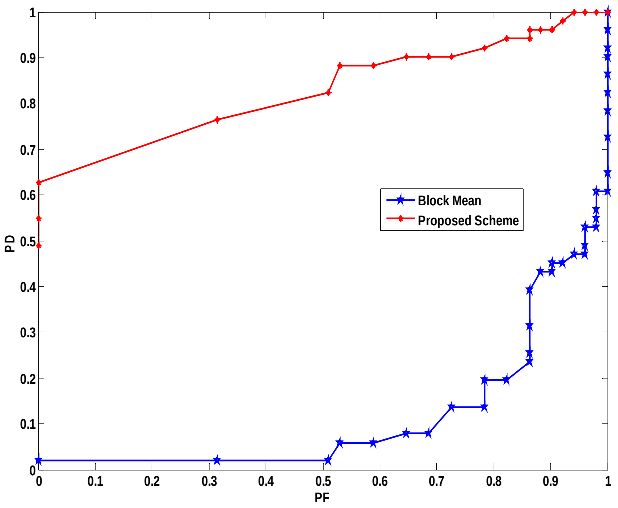


Fig. 11 ROC curve for the Median Filtering Effects