International Journal of Scientific & Engineering Research, Volume 6, Issue 4, April-2015 854
ISSN 2229-5518

# A Novel Artifact Reducing Approach Based on Sparse representation on jpeg compressed Fingerprint images

Nisha Elsa Varghese, Gopakumar G, Tinu Elizabeth thomas

**Abstract**— Among the biometric recognition technology, fingerprint recognition is very popular for personal identification due to the uniqueness, universality, collectability and invariance. Large volumes of fingerprint are collected and stored every day in a wide range of applications, including forensics and access control. Fingerprint image compression is a key technique to solve this problem. Among the compression techniques, JPEG compression is widely used for fingerprint compression, since it provides good compressed results. However, the standard JPEG decompression result usually contains some artifacts, such as blocking artifacts. Compression artifacts are a noticeable distortion of media caused by lossy compression technique. This artifact usually occurs when the compression ratio is high. So we can use the method of sparse representation via k-svd in jpeg compression to remove blocking artifacts in jpeg compressed fingerprint images. The blocking artifacts removal can be measured in terms of psnr and ssim vaues. The experimental results show that the psnr and ssim values shows significant improvement after k-svd algorithm in jpeg compressed image. Also watermarking in spatial domain can be applied to artifact removed fingerprint image to enhance authenticity. Matlab R2012b is used for implementation of the algorithm.

**Index Terms**— Artifacts , Fingerprint, JPEG compression, k-svd, , PSNR ,sparse representation,SSIM, watermarking.

———————————— ◆ ————————————

## 1 Introduction

With widespread application of fingerprint recognition in image forensics and access control ,large volumes of fingerprints are collected and stored every day. This consumes a large amount of memory. Fingerprint image compression is a key technique to solve the problem. A new fingerprint compression algorithm based on sparse representation is introduced as in[1]. Generally, compression technologies can be classed into lossless and lossy. Lossless compression allows the exact original images to be reconstructed from the compressed data. Lossy compression technologies usually transform an image into another domain, quantize and encode its coefficients as in [1].

Transform based image compression can be classified into two types. Two most common options of transformation are the Discrete Cosine Transform (DCT) [2] and the Discrete Wavelet Transform (DWT) [3].The DCT-based encoder can be thought of as compression of a stream of 8 X 8 small blocks of images. This transform has been adopted in JPEG [4]. The JPEG compression scheme has many advantages such as simplicity, universality and availability. However, it has a bad performance at low bit-rates mainly because of the underlying block-based DCT scheme.

JPEG compression [5 ] technique usually consists of three stages. We can apply jpeg compression to the input fingerprint image .The first step is to divide this input fingerprint image into non-overlapping block of size 8x8. Next step is to apply forward discrete cosine transform (DCT) on each block. The output of DCT is frequency domain or certain DCT coefficients. The second stage is divide the above DCT coefficient by Quantization table and rounded to nearest integers. Final stage is applying lossless compression (Entropy encoding) to produce the compressed result. Fig 1 shows the standard JPEG compression and Decompression schemes.

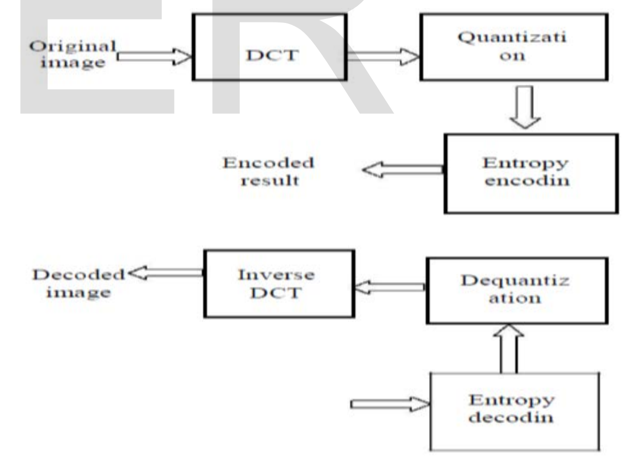



Fig. 1.*UpperDiagram: JPEG Compression; Lower Diagram: JPEG Decompression [5]*

The main problem of image compression is that certain round of error will occur at the time of quantization. When compression ratio is high, more visible artifacts appear in decompressed fingerprint images. A compression artifact is a noticeable distortion of media caused by the application of compression. Artifacts can appear when we perform block-based coding for quantization, as in JPEG-compressed fingerprint images.

ISSN 2229-5518

Compression technique divides the original finger print image into sub images are called blocks. Each block is transformed and the selected large coefficients are quantized and then stored/transmitted. The drawback is that the discontinuities at the block boundaries are visible. Such types of artifacts are termed as blocking artifacts. Since fingerprint images can be used for authentication the blocking artifacts in jpeg compressed image is not tolerable.

A novel approach based on sparse representation using k-svd [5] to remove artifacts in jpeg compressed fingerprint image is proposed. Numerical experiments will be performed to demonstrate that the proposed method outperforms jpeg decompression methods in the measure of peak of signal to noise ratio and structural similarity index after artifact removal in fingerprint image.

The remaining part of the paper is organized as follows.The proposed algorithm of artifact removal of fingerprint images is given in Section 2. In Section 3, watermarking on artifact removed fingerprint image is done .In section 4, numerical tests are done to demonstrate the efficiency of the proposed method. We conclude the paper in Section 5.

## 2 PROPOSED ALGORITHM FOR ARTIFACT REMOVAL OF JPEG COMPRESSED FINGERPRINT IMAGES



The various steps of artifact removal are :

1. Jpeg compression on fingerprint image.
2. Dictionary Learning via k-svd [5][6][7].
3. Llyod's Quantization on fingerprint image.
4. Watermarking on artifact removed fingerprint image.

The algorithm for artifact removal on fingerprint images is as follows:

1.The input fingerprint image is compressed using jpeg algorithm.

1.1 Stage 1: is to split the whole image into non-overlapping blocks of size 8 X8, and to apply the discrete cosine transformation (DCT) on each block.

1.2 Stage 2: divide the above cosine transform coefficients by a quantization table point wisely, and the quantized values are rounded to their nearest integers.

2 The jpeg compressed fingerprint image contains artifacts which can be removed by sparse representation[1],[5].

2.1 For a given jpeg compressed fingerprint image ,slice it into small patches.

2.2 For each patch ,its mean is calculated and subtracted from the patch.

2.3 For each patch, solve l0-minimization problem by MP method.

2.4 Those coefficients whose absolute value are less than a given threshold are treated as zero. Record the remaining coefficients and their locations.

3 Encode the atom number of each patch ,the mean value of each patch, and the indexes; quantize and encode the coefficients.

4 Output the artifact removed fingerprint image after Llyod's quantization.

### *2.1 JPEG compression on fingerprint image*

The first step is to apply jpeg compression on the input fingerprint image [5] . DCT is the first step for image compression. It can be performed in 8X8 blocks. The pixel value of black and white image ranges from 0 to 255.Here pure black pixel is represented by 0 and pure white pixel is represented by 1. But DCT is designed to work on pixel ranges from -128 to 127; the original block is leveled off by subtracting 128 from each entry.

Next step is applying DCT in each block results a frequency domain. Our 8X8 DCT coefficients are ready for Quantization. Useful feature of image compression is varying level of image compression and quality is obtained through the selection of specific quantization matrices. Quality level ranges from 1 to 100 where 1 gives poorest image quality and highest compression and 100 gives best quality and lowest compression.

The standard quantization matrix is multiplied by (100-quality level)/50 if quantization level greater than 50. The standard quantization matrix is multiplied by 50/Quality level if quality level less than 50. The resultant scaled quantization matrix is rounded to positive integer rounded from 1 to 255. The Quantization technique is achieved by dividing each element in DCT coefficient by the corresponding element in quantization matrix which is rounded to nearest integers. Next step is entropy encoding to produce the compressed result. For performing quantization, a default JPEG quantization matrix is used.The value of q is chosen to be q=50/3.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 16 | 11 | 10 | 16 | 24 | 40 | 51 | 61 |
| 12 | 12 | 14 | 19 | 26 | 58 | 60 | 55 |
| 14 | 13 | 16 | 24 | 40 | 57 | 69 | 56 |
| 14 | 17 | 22 | 29 | 51 | 87 | 80 | 62 |
| 18 | 22 | 37 | 56 | 68 | 109 | 103 | 77 |
| 24 | 35 | 55 | 64 | 81 | 104 | 113 | 92 |
| 49 | 64 | 78 | 87 | 103 | 121 | 120 | 101 |
| 72 | 92 | 95 | 98 | 112 | 100 | 103 | 99 |

*Fig 2: Default quantization matrix*

ISSN 2229-5518

### 2.2 *Dictionary Learning via k-svd.*

The jpeg compressed fingerprint image contains blocking artifacts, which can be removed via sparse representation using k-svd algorithm as in [6],[7].

A fixed dictionary is learned via K-SVD as in [5],[6],[7], and the restored images are then derived based on the learned dictionary. It is a two-step iterative method.The first step is to use the orthonormal matching pursuit (OMP)as in [1],[8]algorithm to update the encoding coefficients. The second step is to use SVD to update the dictionary. The combined equation used for dictionary learning is given below:

$$\left(\gamma^*_{i,j}, D^*\right) = \arg\min_{\gamma_{i,j}, D} \sum_{i,j \in P} \left(\frac{1}{2}\|R_{i,j}u^0 - D\gamma_{i,j}\|_2^2 + \mu_{i,j}\|\gamma_{i,j}\|_0\right)$$

The combination of the first and second terms requires the restored image is a sparse linear combination of elements in the dictionary. The advantage of this model is that we determine a sparse representation of the restored image in the learned dictionary so that the restored image can keep the features of the original input image. $\left(\gamma^*_{i,j}, D^*\right)$ encoding coefficients and dictionary determined in the above step. $\|R_{i,j}u^0 - D\gamma_{i,j}\|_2^2$ is related to the representation of their stored image in the dictionary as in [5].

The term $\|\gamma_{i,j}\|_0$ is used to require the encoding coefficients vector to be sparse. Fig 3(a) shows the compressed fingerprint image and Fig 3(b) is the learned dictionary by using KSVD method.Here we are finding the sparse representation from the compressed image.

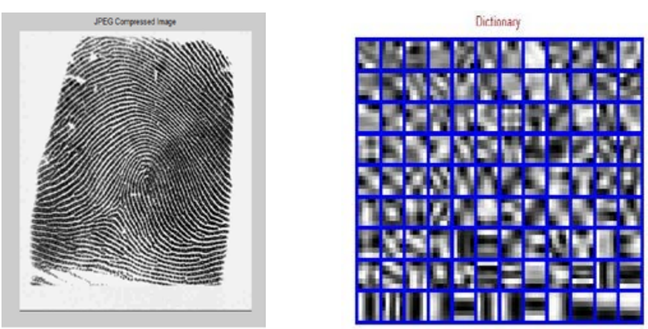



Fig. 3. *(a) Compressed fingerprint image (b) Dictionary learned by K-SVD*

The K-SVD algorithm cannot be used on large images, so the fingerprint image should be divided into small patches first[5]. We divide the 512X512 image into many 12×12image patches. We organize these patches as column vectors. We shall assume that these patches follow the sparse-land model. These small patches are denoised and then tiled to form the reconstructed image. In doing so, visible artifacts may occur on block boundaries. In order to prevent such blockiness artifacts, we propose to work on overlapping patches and average the results [9].

### 2.3 *Llyod's quantization on fingerprint image*

Quantization is the procedure of constraining something from a continuous set of values (such as the real numbers) to a discrete set (such as the integers). The quantization of coefficients is performed using the Lloyd algorithm [10], learnt off-line from the coefficients which are obtained from the training set by the MP algorithm over the dictionary.

The Lloyds algorithm, also known as Voronoi iteration or relaxation, is an algorithm for grouping data points into a given number of categories, used for k-means clustering. Lloyds algorithmas in [11] is usually used in a Euclidean space, so the distance function serves as a measure of similarity between points, and averaging of each dimension for the averaging, but this need not be the case. Lloyd's algorithm starts by partitioning the input points into k initial sets, either at random or using some heuristic. It then calculates the average point, or centroid, of each set via some metric (usually averaging dimensions in Euclidean space). It constructs a new partition by associating each point with the closest centroid, usually using the Euclidean distance function. Then the centroids are recalculated for the new clusters, and algorithm repeated by alternate application of these two steps until convergence, which is obtained when the points no longer switch clusters (or alternatively centroids are no longer changed).

## 3 *Watermarking on artifact removed fingerprint image*

Watermarking is the process whereby a host media is embedded with data for the purpose of protection and authentication. In general digital watermarking involves two major operations: (i) watermark embedding, and (ii) watermark extraction. The secret message embedded as watermark can almost be anything, for example, a bit string, serial number, plain text, image, etc. The most important properties of any digital watermarking technique are: robustness, security, imperceptibility, complexity, and verification.classified according to the human perception (robust or fragile). In images, the watermarking techniques can broadly be classified into three types: (i) visible watermark, (ii) invisible fragile watermark and (iii) invisible robust watermark, which is widely used .

Three tasks are done by our embedding algorithm as in [12]: generates extraction key, determines the position of the embedding watermark and then embeds watermark.

First of all, the algorithm must determine where the watermark should be located; therefore, a base-image is divided into 64 blocks. Each block size is: (M/8 x N/8) where M and N

ISSN 2229-5518

are the number of the rows and columns of the base image respectively. So there will be eight groups of rows as well as for columns.

The watermark will embed through one of these groups either rows or columns in each base image but randomly for sequence base images. In the selected group a LSB (least significant bit) over all pixels will extract (LSB 0) as in [13].Then MSB of mark image is extracted. Then MSB of mark is inserted in to the LSB of the base image.

### 3.1 Extraction Algorithm

To extract the watermark, the algorithm needs to decrypt extraction key to obtain the initial key, through which the algorithm can locate the watermark position in the targeted base-image. Algorithm needs to divide marked-image to 64 blocks, and then determining the watermark location.

The extraction operation begins from the 1st block of the determined group by extracting the LSB from each pixel in the group, and then is multiplied by the ordering number of the block before adding it to the corresponding pixel in the zero-image (by this step all pixels LSB will become valued not default Zero).This will be repeated eight times to revalue the zero-images pixels with watermark real values obtained from the marked-image LSBs, and the result will be the watermark.

Steps for watermarking.

1. Read Watermark Image and key to encrypt image.
2. Read Base Image-Artifact removed jpeg fingerprint image.
3. Resize Watermark Image.
4. Divide Base Image (8 X 8).
5. Determine the Watermarking Location:
6 .Embed Watermark into Base Image and perform bitwise X-OR operation with the key and watermarked image.
7 .Result is:Invisible Watermark.
8. To de watermark the image, first the watermarked image is bitwise X-oRed with the key and finally we can extract the image.

The base image is artifact removed fingerprint image .The mark image is embedded into base image to get marked image. Finally we can extract the mark image.The watermarking on artifact removed jpeg compressed fingerprint image is as shown in fig 4 as follows:

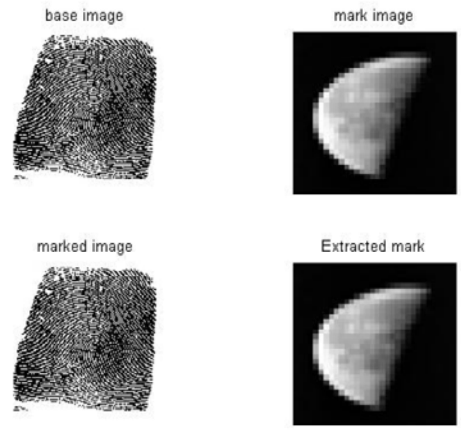



*Fig 4*: *Watermarking on artifact removed fingerprint image*

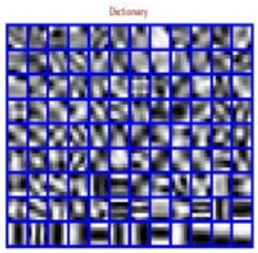
## 4 EXPERIMENTAL RESULTS AND PERFORMANCE EVALUATION

Numerical experiments are performed on three databases namely, Database-1 ,Database-2,Database-3 which can be downloaded from[14] .The size of the fingerprints in the database is 640 X 640[1]. The quality of fingerprints in the database is good. The size of patches for dictionary training is chosen to be 12 X 12 [1].

### *4.1 Experimental Results*

The fig 4 (a) shows original input fingerprint image (b) shows jpeg compressed fingerprint image and (c) shows artifact removed fingerprint image.


Fig 4(a)original input fingerprint image (b) Jpeg compressed fingerprint image.




(c)Dictionary learned from jpeg compressed image (d).Artifact removed fingerprint image

### 4.2 Performance Evaluation

For evaluating the quality of compressed image two parameters are used, which is listed below:-

1. PSNR (Peak signal to noise ratio)value:

The PSNR[5] block computes the peak signal-to-noise ratio, in decibels, between two images. This ratio is often used as a quality measurement between the original and a compressed image. The higher the PSNR, the better the quality of the compressed, or reconstructed image.From this value we can ensure the output is received with the same or good quality at the receiver side.

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right)$$

where MSE is given by,

$$MSE = \sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\frac{[OI(i,j) - DI(i,j)]^2}{MxN}$$

where OI is original image and DI is the distorted image ie compressed image.

2.SSIM(Structural Similarity Index):

The average SSIM index[15] is used to evaluate the overall image quality. The larger the value is, the better the restoration result. The local SSIM index is defined by:

$$SSIM(u,u_r) = \frac{1}{N}\sum_{i=1}^{N} SSIM_{local}\ (u(i),u_r\ (i))$$

where μ and σ are mean and standard deviation respectively.c1,c2 are the stabilizing constants.

SSIM has a value between 0 to 1. Similar images have SSIM near to 1.

$$SSIM_{local}(u(i),u_r(i)) = \frac{[2\mu(u(i)\mu(u_r(i)) + c1][2\sigma(u(i)u_r(i)) + c2]}{[\mu^2(u(i)) + \mu^2(u_r(i)) + c1][\sigma^2(u(i)) + \sigma^2(u_r(i)) + c2]}$$

The experimental results show that the psnr and ssim values has significant improvement after artifact removal of jpeg compressed fingerprint image.The effects of psnr and ssim values before k-svd on fingerprint image and after artifact removal on database 1,2 3 is as follows:

| | JPEG compression | Proposed method |
|---|---|---|
| Databases:input fingerprint images | Psnr, ssim | Psnr, ssim |
| Db1,0-1.bmp | 28.4799,0.9315 | 36.617,0.964 |
| Db2 2-1.bmp | 29.5274,0.9350 | 36.7652,0.9688 |
| Db3 1-1.bmp | 28.4137,0.9201 | 35.8142,0.9484 |

## 5 Conclusion

A novel artifact removal for jpeg compressed fingerprint images is introduced. The jpeg compressed fingerprint images contain artifacts at high compression ratios. So a novel artifact reducing method based on sparse representation is proposed.The experiments reflect that the dictionary obtained by the K-SVD algorithm works best. Moreover, the larger the number of the training set is, the better the compression result is. One of the main difficulties in developing compression algorithms for fingerprints resides in the need for preserving the minutiae which are used in the identification. The experiments show that our algorithm can hold most of the minutiae robustly during the compression and reconstruction. The experimental results show that the psnr and ssim values shows significant improvement after ksvd algorithm in jpeg compressed image.The jpeg compressed artifact removed fingerprint image can also be used in applications like watermarking.As a future work, the optimization algorithms for solving the sparse representation need to be investigated.

## 6 References

[1]. Guangqi Shao, Yanping Wu, Yong A, Xiao Liu, and Tiande Guo,"Fingerprint Compression Based on Sparse Representation ", IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 23, NO. 2, FEBRUARY 2014.

[2] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," IEEE Trans. Comput., vol. C-23, no. 1, pp. 90–93, Jan. 1974.

ISSN 2229-5518
[3] C. S. Burrus, R. A. Gopinath, and H. Guo, Introduction to Wavelets and Wavelet Transforms: A Primer. Upper Saddle River, NJ, USA: Prentice-Hall, 1998.

[4]W. Pennebaker and J. Mitchell, "*JPEG—Still Image Compression Standard"*,New York, NY, USA: Van Nostrand Reinhold, 1993.

[5] Huibin Chang, Michael K.Ng, Tieyong Zeng "Reducing Artifacts in JPEG Decompression via a learned dictionary", IEEE transaction on signal processing Vol.62,No 3,Februvary 2014.

[6] M. Aharon, M. Elad, and A. M. Bruckstein, "On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them," J. Linear Algebra Appl., vol. 416, no. 1, pp. 48–67, 2006.

[7] M. Aharon, M. Elad, and A. M. Bruckstein, "The K-SVD: An algorithm for designing of over complete dictionaries for sparse representation,"IEEE Trans. Signal Process., vol. 54, pp. 4311–4322, 2006.

[8] Matthew C. Stamm, Steven K. Tjoa, W. Sabrina Lin, and K. J. Ray Liu, "AntiForensics of JPEG compression", in Proc. Int. Conf. Acoustics, Speech, and Signal Processing, Dallas, TX, Apr. 2010.

[9].Michael Elad and Michal Aharon" Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries", IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 15, NO. 12, DECEMBER 2006

[10]S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.

[11] Martin Mayer, "Quantization of Images and Lloyd's Algorithm" ,389.130 ,Bachelor Thesis Institute of Communications and Radio-Frequency Engineering ,September 26, 2010,

[12] Mustafa Osman Ali, Elamir Abu Abaida Ali Osman, Rameshwar Row,Electronics & Communication Engineering Dept., Biomedical Engineering Dept. University College of Engineering, Osmania University, Hyderabad, India."Invisible Digital Image Watermarking in Spatial Domain",International Journal of Engineering and Innovative Technology (IJEIT) Volume 2, Issue 5, November 2012.

[13] S Jayaraman, S Esakkirajan, and T Veerakumar: Digital Image Processing. McGraw-Hill, 2009.

[14] (2013, Jun.). *Fingerprint Images* [Online]. Available: http://pan.baidu.com/share/link?shareid=3872475108&uk=2301599420

[15] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.